Systems Analysis Programs for Hands-On Integrated Reliability Evaluations (SAPHIRE) Version 8

Volume 1 Overview and Summary

C. L. Smith
S. T. Wood

March 2011

INL
Idaho National Laboratory

Systems Analysis Programs for Hands-On Integrated Reliability Evaluations (SAPHIRE) Version 8

Volume 1 Overview and Summary

C. L. Smith
S. T. Wood

March 2011

Idaho National Laboratory
Idaho Falls, Idaho 83415

http://www.inl.gov

AVAILABILITY NOTICE

Availability of Reference Materials Cited in NRC Publications

Most documents cited in NRC publications will be available from one of the following sources:

1.   The NRC Public Document Room, Rockville Pike, Rockville, MD 20852 (pdr@nrc.gov)

2.   The Superintendent of Documents, U. S. Government Printing Office (GPO), Mail Stop SSOP, Washington, DC 20402-9328

3.   The National Technical Information Service, Springfield, VA 22161

Although the listing that follows represents the majority of documents cited in NRC publications, it is not intended to be exhaustive.

Referenced documents available for inspection and copying for a fee from the NRC Public Document Room include NRC correspondence and internal NRC memoranda; NRC bulletins, circulars, information notices, inspection and investigative notices; licensee event reports; vendor reports and correspondence; Commission papers; and applicant and licensee documents and correspondence.

The following documents in the NUREG series are available for purchase from the GPO Sales Program: formal NRC staff and contractor reports, NRC-sponsored conference proceedings, international agreement reports, grant publications, and NRC booklets and brochures. Also available are regulatory guides, NRC regulations in the *Code of Federal Regulations*, *and Nuclear Regulatory Commission Issuances*.

Documents available from the National Technical Information Service include NUREG-series reports and technical reports prepared by other Federal agencies and reports prepared by the Atomic Energy Commission, forerunner agency to the Nuclear Regulatory Commission.

Documents available from public and special technical libraries include all open literature items, such as books, journal articles, and transactions. *Federal Register* notices, Federal and State legislation, and congressional reports can usually be obtained from these libraries.

Documents such as theses, dissertations, foreign reports and translations, and non-NRC conference proceedings are available for purchase from the organization sponsoring the publication cited.

Single copies of NRC draft reports are available free, to the extent of supply, upon written request to the Office of Administration, Distribution and Mail Services Section U. S. Nuclear Regulatory Commission, Washington, DC 20555-0001.

The public maintains copies of industry codes and standards used in a substantive manner in the NRC regulatory process at the NRC Library, Two White Flint North, 11545 Rockville Pike, Rockville, MD, 20852, for use. Codes and standards are usually copyrighted and may be purchased from the originating organization or, if they are American National Standards, from the American National Standards Institute, 1430 Broadway, New York, NY 10018.

PREVIOUS REPORTS

S. T. Wood, C. L. Smith, K. J. Kvarfordt, S. T. Beck, *Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE) Vol. 1 Summary Manual*, NUREG/CR-6952, August 2008.

C. L. Smith, S. T. Wood, W. J. Galyean, J. A. Schroeder, S. T. Beck, M. B. Sattison, *Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE) Vol. 2 Technical Reference*, NUREG/CR-6952, August 2008.

K. J. Kvarfordt, S. T. Wood, C. L. Smith, *Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE) Vol. 3 Code Reference Manual*, NUREG/CR-6952, August 2008.

S. T. Beck, S. T. Wood, C. L. Smith, *Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE) Vol. 4 Tutorial*, NUREG/CR-6952, August 2008.

C. L. Smith, J. Schroeder, S. T. Beck, *Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE) Vol. 5 GEM Manual*, NUREG/CR-6952, August 2008.

C. L. Smith, R. Nims, K. J. Kvarfordt, C. Wharton, *Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE) Vol. 6 Quality Assurance Manual*, NUREG/CR-6952, August 2008.

K. J. Kvarfordt, S. T. Wood, C. L. Smith, *Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE) Vol. 7 Data Loading Manual*, NUREG/CR-6952, August 2008.

Smith, C. L., et al., *Testing, Verifying, and Validating SAPHIRE Versions 6.0 and 7.0*, NUREG/CR-6688, October 2000.

K. D. Russell, et al. *Systems Analysis Programs for Hands-on Reliability Evaluations (SAPHIRE) Version 6.0 - System Overview Manual*, NUREG/CR-6532, May 1999.

K. D. Russell et al., *Integrated Reliability and Risk Analysis System (IRRAS) Version 5.0, Volume 2 - Reference Manual*, NUREG/CR-6116, EGG-2716, July 1994.

K. D. Russell et al., *Verification and Validation (V&V), Volume 9 – Reference Manual*, NUREG/CR-6116, EGG-2716, July 1994.

K. D. Russell et al., *Integrated Reliability and Risk Analysis System (IRRAS) Version 4.0, Volume 1 - Reference Manual*, NUREG/CR-5813, EGG-2664, January 1992.

K. D. Russell et al., *Integrated Reliability and Risk Analysis System (IRRAS) Version 2.5 Reference Manual*, NUREG/CR-5300, EGG-2613, March 1991.

K. D. Russell, M. B. Sattison, D. M. Rasmuson, *Integrated Reliability and Risk Analysis System (IRRAS) - Version 2.0 User's Guide*, NUREG/CR-5111, EGG-2535, manuscript completed March 1989, published June 1990.

K. D. Russell, D. M. Snider, M. B. Sattison, H. D. Stewart, S.D. Matthews, K. L. Wagner, *Integrated Reliability and Risk Analysis System (IRRAS) User's Guide - Version 1.0 (DRAFT)*, NUREG/CR-4844, EGG-2495, June 1987.

ABSTRACT

The Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE) is a software application developed for performing a complete probabilistic risk assessment (PRA) using a personal computer (PC) running the Microsoft Windows operating system. SAPHIRE Version 8 is funded by the U.S. Nuclear Regulatory Commission (NRC) and developed by the Idaho National Laboratory (INL). INL's primary role in this project is that of software developer and tester. However, INL also plays an important role in technology transfer by interfacing and supporting SAPHIRE users, who constitute a wide range of PRA practitioners from the NRC, national laboratories, the private sector, and foreign countries.

SAPHIRE can be used to model a complex system's response to initiating events and quantify associated consequential outcome frequencies (or probabilities). Specifically, for nuclear power plant applications, SAPHIRE 8 can identify important contributors to core damage (Level 1 PRA) and containment failure during a severe accident which leads to releases (Level 2 PRA). It can be used for a PRA where the reactor is at full power, low power, or at shutdown conditions. Furthermore, it can be used to analyze both internal and external initiating events and has special features for managing models such as flooding and fire. It can also be used in a limited manner to quantify risk, using PRA techniques, in terms of release consequences to the public and environment (Level 3 PRA).

In SAPHIRE 8, the act of creating a model has been separated from the analysis of that model in order to improve the quality of both the model (e.g., by avoiding inadvertent changes) and the analysis. Consequently, in SAPHIRE 8, the analysis of models is performed by using what are called Workspaces. Currently, there are Workspaces for three types of analyses: (1) the NRC's Accident Sequence Precursor program, where the workspace is called "Events and Condition Assessment (ECA);" (2) the NRC's Significance Determination Process (SDP); and (3) the General Analysis (GA) workspace. Workspaces are independent of each other and modifications or calculations made within one workspace will not affect another. In addition, each workspace has a user interface and reports tailored for their intended uses.

This report provides an overview of the functions and features available in SAPHIRE 8 and presents general instructions for using the software. Since SAPHIRE 8 expands upon Version 7, new and improved features will be discussed.

FOREWORD

The U.S. Nuclear Regulatory Commission (NRC) has developed the Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE) software that is used to perform probabilistic risk assessments (PRAs) on a personal computer. SAPHIRE enables users to supply basic event data, create and solve fault and event trees, perform uncertainty analyses, and generate reports. In that way, analysts can perform PRAs for any complex system, facility, or process.

For nuclear power plant PRAs, SAPHIRE can be used to model a plant's response to initiating events, quantify core damage frequencies, and identify important contributors to core damage (Level 1 PRA). The program also can be used to evaluate containment failure and release models for severe accident conditions given that core damage has occurred (Level 2 PRA). In so doing, the analyst could build the PRA model assuming that the reactor is initially at full power, low power, or shutdown. In addition, SAPHIRE can be used to analyze both internal and external events and, in a limited manner, to quantify the frequency of release consequences (Level 3 PRA). Because this software is a very detailed technical tool, users should be familiar with PRA concepts and methods used to perform such analyses.

SAPHIRE has evolved with advances in computer technology and users' needs. Starting with Version 5, SAPHIRE operated in the Microsoft Windows™ environment. Versions 6 and 7 included features and capabilities for developing and using larger, more complex models. SAPHIRE Version 8 includes significant new features and capabilities to meet user needs for NRC risk-informed programs. In general, these include:

- Improved user interfaces supporting NRC's Significance Determination Process, event and condition assessments, and more detailed types of PRA analyses.

- Development and use of NRC's Standardized Plant Analysis Risk models.

- New and improved solving algorithms.

- Support features for user-friendliness.

This NUREG-series report comprises seven volumes as outlined below and incorporates new features and capabilities of Version 8.

Volume 1, "Overview and Summary"

Volume 1 provides an overview of the functions and features available in SAPHIRE Version 8 and presents general instructions for using the software.

Volume 2, "Technical Reference"

Volume 2 summarizes the fundamental mathematical concepts of sets and logic, fault trees, and probability. It then describes the algorithms used to construct a fault tree and to obtain the minimal cut sets. This report presents the formulas used to obtain the probability of the top event from the minimal cut sets and the formulas for probabilities that apply for various assumptions concerning reparability and mission time. In addition, it defines the measures of basic event importance that SAPHIRE can calculate. This volume also gives an overview of uncertainty analysis using simple Monte Carlo sampling or Latin Hypercube sampling and states the algorithms used by this program to generate random basic event probabilities from various distributions. Finally, this report

discusses enhanced and new capabilities such as post-processing rules, integrated model solving using model types, and workspace analysis routines.

Volume 3, "Users' Guide"

Volume 3 provides a brief discussion of the purpose and history of the software as well as general information such as installation instructions, starting and stopping the program, and some pointers on how to get around inside the program. Next, it discusses database concepts and structure. The following nine sections (one for each of the menu options on the SAPHIRE main menu) furnish the purpose and general capabilities for each option. Finally, Volume 3 provides the capabilities and limitations of the software.

Volume 4, "Tutorial"

Volume 4 provides a series of lessons that guide the user through basic steps common to most analyses performed with SAPHIRE.

Volume 5, "Workspaces"

Volume 5 describes the functionality and process behind SAPHIRE Version 8 workspaces. Workspaces provide an area in which a PRA model can be analyzed to obtain risk insights for a given initiating event or condition. Workspaces replace the "Graphical Evaluation Module" in earlier SAPHIRE versions.

Volume 6, "Quality Assurance"

Volume 6 is designed to describe how the SAPHIRE software quality assurance (QA) is performed for Version 8, what constitutes its parts, and the limitations of those processes. In addition, this report describes the Independent Verification and Validation that was conducted for Version 8 as part of an overall QA process.

Volume 7, "Data Loading"

Volume 7 is designed to guide the user through the basic procedures necessary to enter PRA data into the SAPHIRE program using SAPHIRE's MAR-D ASCII-text (or "flat file") data formats. In addition, this manual covers loading data through the new Accident Sequence Matrix and discusses the Project Integrate interfaces with SAPHIRE.

<div align="right">

Christiana H. Lui, Director

Division of Risk Analysis

Office of Nuclear Regulatory Research

U.S. Nuclear Regulatory Commission

</div>

CONTENTS

LIST OF FIGURES

LIST OF TABLES

EXECUTIVE SUMMARY

The U.S. Nuclear Regulatory Commission (NRC) has developed a powerful personal computer (PC) software application for performing probabilistic risk assessments (PRAs), called Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE).   Using SAPHIRE 8 on a PC, an analyst can perform a PRA for any complex system, facility, or process. Regarding nuclear power plants, SAPHIRE can be used to model a plant's response to initiating events, quantify associated core damage frequencies and identify important contributors to core damage (Level 1 PRA). It can also be used to evaluate containment failure and release models for severe accident conditions, given that core damage has occurred (Level 2 PRA). It can be used for a PRA assuming that the reactor is at full power, at low power, or at shutdown conditions. Furthermore, it can be used to analyze both internal and external initiating events, and it has special features for transforming models built for internal event analysis to models for external event analysis.

SAPHIRE development and maintenance has been undertaken by the Idaho National Laboratory (INL). The INL began development of a PRA software application on a PC in the mid 1980s when the potential of PC applications started being recognized. The initial version, Integrated Risk and Reliability Analysis System (IRRAS), was released by the Idaho National Engineering Laboratory (now Idaho National Laboratory) in February 1987. IRRAS was an immediate success, because it clearly demonstrated the feasibility of performing reliability and risk assessments on a PC and because of its tremendous need (Russell 1987). The development of IRRAS continued over the following years, leading SAPHIRE to become a powerful PRA tool.

SAPHIRE 8, building upon earlier versions, provides the functions required for performing a PRA. Users can supply basic event data, create and solve fault trees and event trees, perform uncertainty analyses, and generate reports. Extending this analysis, Workspaces are user-friendly interface that streamlines and automates select SAPHIRE analysis processes.

SAPHIRE contains improved editors or options for creating event trees and fault trees, defining accident sequences and basic event failure data, solving system fault trees and accident sequence event trees, quantifying cut sets, performing sensitivity and uncertainty analyses, documenting the results, and generating reports. SAPHIRE capabilities for performing a PRA are summarized below.

- Graphical fault tree construction
- Graphical event tree construction
- Rule-based fault tree linking
- Fast cut set generation
- Fault tree flag sets
- Failure data
- Uncertainty analysis
- Cut set editor, slice, display, and recovery analysis tools
- Cut set path tracing
- Cut set comparison
- Cut set post-processing rules
- Cut set end state partitioning
- End state analysis

- User-defined model types
- User-defined basic event attributes

SAPHIRE 8 has a powerful report generation module. In addition, SAPHIRE has utility functions to recover a corrupted database, check project information for quality purposes, and change user-defined constants. With SAPHIRE, a user can export the graphical fault trees and event trees to an industry-standard Windows metafile or JPEG files.

To use SAPHIRE, the user must create a project. A project is any logical grouping of fault trees and sequences with their associated basic events, cut sets, reliability data, and descriptions. Along with the project name, description, plant design, etc., it also contains cross-reference information for relationships between basic events, fault tree, sequence, event tree, end state, change sets, and gate in the project. The project also contains the project sequence partition, and project sequence, fault tree, and end state recovery information. Once the project is created, the user can construct fault trees and event trees.

SAPHIRE Version 8 replaces the Version 7. Changes to SAPHIRE features and capabilities include:

- Improved and stream-lined user interfaces
- Enhanced model development
- New analysis methods
- Revamped general support features

SAPHIRE Version 8 is designed to easily handle larger and more complex models to be used. Applications of previous versions indicated the need to build and solve models with a large number of sequences.   In addition, the complexity of the models has increased since risk assessments evaluate both potential internal and external event initiators, as well as different plant operational states in which the initiating event may occur. Special features have been designed in SAPHIRE Version 8 to help create and run integrated models which may be composed of a number of different model types (e.g., models with different types of initiating events or plant operational states).   External events models can be built more expeditiously through the use of automation tools. Any combination of model types can be solved, and a powerful graphical editor allows examination of the underlying logic.

New modeling and calculation methods have also been implemented. For example, phase mission time analysis capability has been implemented. This capability was incorporated in support of the NRC's "extended Level 1" and large, early, release frequency (LERF)/Level 2 standardized plant analysis risk (SPAR) models; however, it may also be useful for low power and shutdown models, which may consider multiple plant operational states. For common-cause failure (CCF) modeling, the Risk Assessment Standardization Project methodology has been incorporated. Also, CCF probabilities are now automatically adjusted to account for the impact of sequence flag sets. In addition, SAPHIRE Version 8 offers an improved sequence solving algorithm which addresses limitations in the current solving algorithm related to application of sequence recovery rules.

The uncertainty analysis functions in SAPHIRE estimate the variability (due to the uncertainties in the basic event probabilities) of a fault tree top event probability, an event tree sequence frequency, and end state frequency, or any of the importance measures. In an uncertainty analysis, SAPHIRE samples the user-specified distributions for each basic event in a group of cut sets, then quantifies these cut sets using the sampled values.

Finally, the functionality, in general, that was available in Version 7 is available in Version 8, but in an improved manner. One of the strengths of SAPHIRE lies in its computation capabilities. Version 8 not only taps into these capabilities, but also is better positioned to leverage these capabilities for non-expert users via an improved graphical user interface. SAPHIRE has become a powerful and easy to use PRA tool. Its relational database structure and editing rules offer the capability for sophisticated modeling of accident progression and, therefore, offer the means for a more accurate and efficient analysis. Several other features, many constructed from feedback by users dealing with large-scale PRA models, make SAPHIRE among the fastest and most sophisticated PRA codes available today.

ACKNOWLEDGEMENTS

ACRONYMS

ASP	Accident Sequence Precursor
BET	bridge event tree
CET	containment event tree
CCDP	conditional core damage probability
CD	core damage
CDP	core damage probability
CPET	containment phenomena event trees
ECA	Events and Condition Assessment
ETG	event tree graphic
FEP	Fault Tree, Event Tree, and Piping and Instrumentation Diagram
FV	Fussell-Vesely
GEM	Graphical Evaluation Module
INEL	Idaho National Engineering Laboratory
INL	Idaho National Laboratory
IRRAS	Integrated Reliability and Risk Analysis System
LERF	large early release frequency
MAR-D	Models and Results Database
MGL	multiple Greek letter
NRC	Nuclear Regulatory Commission
P&ID	Piping and Instrumentation Diagram
PC	personal computer
PDS	Plant Damage State
PRA	probabilistic risk analysis
QA	Quality Assurance
RASP	Risk Assessment Standardization Project
SA	standard analysis
SAPHIRE	Systems Analysis Programs for Hands-on Integrated Reliability Evaluations
SARA	System Analysis and Risk Assessment
SDP	Significance Determination Process
SPAR	Standardized Plant Analysis Risk
T&M	Test and Maintenance

Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE) Version 8

Volume 1 Overview and Summary

1. INTRODUCTION

1.1 Background

The U.S. Nuclear Regulatory Commission (NRC) has developed a powerful personal computer (PC) software application for performing probabilistic risk assessments (PRAs), called Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE) Version 8.

Using SAPHIRE 8 on a PC, an analyst can perform a PRA for any complex system, facility, or process. Regarding nuclear power plants, SAPHIRE can be used to model a plant's response to initiating events, quantify associated core damage frequencies, and identify important contributors to core damage (Level 1 PRA). It can also be used to evaluate containment failure and release models for severe accident conditions, given that core damage has occurred (Level 2 PRA). It can be used for a PRA assuming that the reactor is at full power, at low power, or at shutdown conditions. Furthermore, it can be used to analyze both internal and external initiating events, and it has special features for transforming models built for internal event analysis to models for external event analysis. It can also be used in a limited manner to quantify risk for release consequences to both the public and the environment (Level 3 PRA). For all of these models, SAPHIRE can evaluate the uncertainty inherent in the probabilistic models.

SAPHIRE development and maintenance has been undertaken by the Idaho National Laboratory (INL). The INL began development of a PRA software application on a PC in the mid 1980s when the enormous potential of PC applications started being recognized. The initial version, Integrated Risk and Reliability Analysis System (IRRAS), was released by the Idaho National Engineering Laboratory (now Idaho National Laboratory) in February 1987. IRRAS was an immediate success, because it clearly demonstrated the feasibility of performing reliability and risk assessments on a PC and because of its tremendous need (Russell 1987). Development of IRRAS continued over the following years. However, limitations to the state of the-art during those initial stages led to the development of several independent modules to complement IRRAS capabilities (Russell 1990; 1991; 1992; 1994). These modules were known as Models and Results Database (MAR-D), System Analysis and Risk Assessment (SARA), and Fault Tree, Event Tree, and Piping and Instrumentation Diagram (FEP).

IRRAS was developed primarily for performing a Level 1 PRA. It contained functions for creating event trees and fault trees, defining accident sequences and basic event failure data, solving system fault trees and accident sequence event trees, quantifying cut sets, performing sensitivity and uncertainty analyses, documenting the results, and generating reports.

MAR-D provided the means for loading and unloading PRA data from the IRRAS relational database. MAR-D used a simple ASCII data format. This format allowed interchange of data

between PRAs performed with different types of software; data of PRAs performed by different codes could be converted into the data format appropriate for IRRAS, and vice-versa.

SARA provided the capability to access PRA data and results (descriptive facility information, failure data, event trees, fault trees, plant system model diagrams, and dominant accident sequences) stored in MAR-D. With SARA, a user could review and compare results of existing PRAs. It also provided the capability for performing limited sensitivity analyses. SARA was intended to provide easier access to PRA results to users that did not have the level of sophistication required to use IRRAS.

FEP provided common access to the suite of graphical editors. The fault tree and event tree editors were accessible through FEP as well as through IRRAS, whereas the piping and instrumentation diagram (P&ID) editor was only accessible through FEP. With these editors an analyst could construct from scratch as well as modify fault tree, event tree, and plant drawing graphical figures needed in a PRA.

Previous versions of SAPHIRE consisted of the suite of these modules. Taking advantage of the Windows 95 (or Windows NT) environment, all of these modules were integrated into SAPHIRE Version 6; more features were added; and the user interface was simplified.

This report provides an overview of the functions and features available in SAPHIRE 8 and presents general instructions for using the software. Since SAPHIRE 8 expands upon Version 7, new and improved features will be discussed.

1.2   Purpose

This report introduces and summarizes this NUREG/CR series, Volumes 1-7, documents that describe the functions of SAPHIRE 8—technical documents that describe why it functions the way it does and user's guides that describe how to use the individual functions. Detailed documentation and instructions for using the code available in the on-line help manual have been reformatted to present in this set of NUREG/CR volumes.

In the past, detailed documentation was provided in hard copy format. A series of NUREG/CR reports was released documenting each individual module discussed above (NUREG/CR-6952; Volumes 1 through 7, 2008), document Version 6 and 7 of the SAPHIRE suite of codes. However, in response to continually increasing NRC needs for PRA capability and in response to feedback from users, SAPHIRE has undergone changes and enhancements. The consequent need for a continual update of detailed SAPHIRE documentation is met by the electronic on-line manual, which is updated with each major release of a SAPHIRE version. This new set of NUREG/CR reports described in this report reflects an updated snapshot of the documentation for the current release of SAPHIRE 8.

1.3   SAPHIRE Manual Overview

The following volumes that constitute this series describe SAPHIRE and its capabilities for performing a PRA. A brief summary of each volume follows below. More detailed descriptions can be found in the respective individual volumes.

- Overview and Summary (Volume 1)

    This volume summarizes this NUREG series and overviews the functions of SAPHIRE.

- Technical Reference (Volume 2)

    This volume presents the theoretical background behind the functions of SAPHIRE and includes the following topics: Set Theoretic Concepts, Review of Fault Tree Logic, Probability Concepts, Determination of Minimal Cut Sets, Quantification Tools for Probabilities and Frequencies, Event Probability Calculation Types, Importance Measures, Uncertainty, and Seismic Calculations.

    This document describes the various analysis features, including the common cause failure plug-ins. In addition, sections describe the process of end state gathering and cut set partitioning and the cut set recovery process for both fault trees and sequences.

- SAPHIRE Users' Guide (Volume 3)

    This manual gives guidance on installation instructions and explains the basic features of SAPHIRE. It is intended to assist in learning how SAPHIRE works and provides a step-by-step approach to using the features of SAPHIRE 8.

- SAPHIRE Tutorial (Volume 4)

    This manual describes the overall process of constructing a PRA database using a simple modeling domain, that of a trucking accident. Many of the SAPHIRE's features are discussed as part of this PRA modeling. In addition, a couple of the "advanced" features in SAPHIRE are used, for example cut set recovery rules and template events.

- Workspaces (Volume 5)

    This manual describes the use of the three types of Workspaces that are provided in SAPHIRE 8.

- SAPHIRE Quality Assurance (QA) (Volume 6)

    This manual details the methods and tests used to provide QA. Configuration control is discussed and testing examples are provided.

- SAPHIRE Data Loading (Volume 7)

    This manual is currently divided into four chapters and four appendices. The first two chapters introduce SAPHIRE and important database concepts. The third chapter discusses a sample database, and the fourth chapter discusses the complete process of loading the sample database. Each of these chapters and appendices are updated and expanded as appropriate. Data conversion methods and issues, including validation, are discussed.

This volume focuses on primarily new features. For details on general use of SAPHIRE, Volumes 3, 4, and 5 provide step-by-step guidance of the software.

1.4   Overall SAPHIRE Features and Capabilities

SAPHIRE features and capabilities for performing a PRA are summarized below. While the list discussed below is not an exhaustive list of all features, it provides a description of SAPHIRE basic capabilities.

- Basic Events

  Basic events provide SAPHIRE with probabilistic information needed to quantity the PRA model. Basic events are logically linked together in fault trees.

  Basic events show the failure data input, type of probability calculation, and plug-in (described below) method selected (e.g., common-cause failure method). The basic event calculation types include user-specified probability, operating component with no repair, operating component with repair, standby component, etc.

- Fault trees

  In SAPHIRE, a fault tree generally represents a failure model.   A fault tree model consists of a top event (usually defined by a heading in an event tree) and a connecting logic structure that models the combinations of basic events that must take place to result in the undesired top event. Thus, all the basic events in the fault tree represent failures, whether they are equipment failures, human errors, or adverse conditions that can contribute to failure of the modeled event. Successful events (those things that should happen) that can contribute to failure of the top event can be included in the fault tree also.

- Event trees

  In SAPHIRE, accident sequences are defined using an event tree to indicate the failure or success of top events. Each top event in the event tree is associated with a system fault tree. Event trees are constructed and modified with a graphical editor, which allows linkage of multiple event trees and creation of very large event trees.

- Fault tree and event tree "drag and drop" editor

  Objects can now be created or integrated in an existing object by dragging and dropping on the graphical editor.

- Flag sets

  Flag sets are a special type of change set. Flag sets are used to modify particular events on a sequence-by-sequence basis. They can change a basic event calculation type to "True," "False," or "Ignore," or the process type. They cannot change the basic event failure probability. Flag sets can be assigned to specific sequences directly. Dynamic flag sets can be assigned based on event tree rules. Flag sets can also affect

common-cause failure basic events if the redundancy level represented in the basic event is compromised by failures modeled in the flag set.

- Rule-based fault tree linking

  Accident sequences are generated using a rule-based fault tree linking module that uses a powerful set of user-defined rules to specify top-event substitutions. These rules allow testing the current state of the solution process at each branch point in the event tree.

- Cut sets

  To solve an accident sequence, SAPHIRE constructs a fault tree for those systems that are defined to be failed in the sequence logic by creating a temporary AND gate with these systems as inputs. SAPHIRE then solves this fault tree using the specified truncation values. This process results in a list of cut sets for the failed systems in the accident sequence. SAPHIRE then uses the "delete term" technique to further reduce this list of failed-system cut sets. The minimal cut sets for the sequence remain after the successful-system cut sets terms are deleted. Default cut set quantification is the minimal cut set upper bound; however, the user may also chose the exact solution or the rare event approximation.

- Uncertainty analysis

  Both Monte Carlo and Latin Hypercube methods are available for performing uncertainty analysis. Uncertainty analysis can be performed on individual sequences or groups of sequences. In addition, uncertainty analysis can be performed on importance measures.

- Importance measures

  The traditional importance measures such as Fussell-Vesely, Risk Increase Ratio or Interval, Risk Reduction Ratio or Interval, and Birnbaum are available. However, because these measures are not suitable for all types of analyses, the NRC developed extended importance measures for passive components analysis that include cumulative Fussell-Vesely, Fussell-Vesely/Birnbaum, and group importance. For some analyses, an importance defined as the conditional core damage probability minus the core damage probability is also calculated. Uncertainty analyses on these measures can be performed using Monte Carlo and Latin Hypercube sampling techniques.

- Cut set editor and recovery analysis tools

  A rule-based editor is available for defining rules for modifying cut sets. These rules can be used to apply recovery on cut sets after an initial analysis has been performed. They can also be used to perform extensive changes to the cut set logic to incorporate new or different conditions.

- Versatile cut set slice and display

  A unique cut set "sort and slicer" is available to search and partition a large number of cut sets in fault trees, sequences, or end states. Multiple sequential slices may be performed.

- Cut set path tracing

  A "path tracer" is available to find the exact path through the logic that generated a particular cut set. This option will trace a cut set from an end state to its source in the logic model.

- Cut set end state partitioning

  Cut sets can be partitioned into "end states" via a set of rules allowing each cut set to be assigned to a specific "end state."

- End state analysis

  An "end state analysis module" provides the ability for gathering and analyzing accident sequence end states.

- Model type analysis

  Model types can be assigned to a basic event's "applicability" property. SAPHIRE provides the ability to define different model types such as random, fire, flood, seismic, etc.

- Phase analysis

  Phases can be specified for each model type to incorporate phase durations. Phase analysis provides the capability to work with end states which follow each other in time on a sequence basis. This capability supports the SPAR LERF/Level 2 models, and may be useful for low power and shutdown models.

- Project Integration and Modification

  Version 8 provides a capability to integrate together large, complex models from separately built model components.

- Multi-Processor Parallel Calculations

  In version 8, multiple objects may be solved (for cut sets) in parallel depending on the availability of multiple computer processors in the PC being used to run SAPHIRE.

- Multiple Object Accessibility

  In version 7, a user could only access one object (e.g., basic events, fault trees, event trees, etc.) at a time. In version 8, multiple objects may be accessed at the same time via multiple windows. A window manager assists in manipulating these objects.

- Plug-in architecture

  A user-callable "plug-in" architecture allows interfacing to a customized library of probability calculation models. For example, shipped with SAPHIRE are plug-ins representing an alpha-factor common cause, a multiple Greek letter common cause, and a "supercomponent." In addition, advanced plug-in modules are available to determine corrosion rates from flow-accelerated corrosion and its corresponding impact on the pipe failure probability.

- Report generation module

  SAPHIRE has a powerful report generation module. Reports are available which are tailored for each workspace, and provide more information and flexibility.

- Search capability

  The global search function will return any object in the project that matches the search such as tree names, gates and basic event names, and including the matched text in the descriptions.

- Embedded Macro Capability

  Macros can be written for SAPHIRE tasks, e.g., solving different model types. For example, an integrated model may contain sequences for both internal and external initiating event models. Using the macros capability, it could be specified that SAPHIRE will solve only internal initiating events models, only external initiating events models, or both types of models. These macros can be assigned user-defined icons and placed on the main toolbar for a streamlined analysis.

- MAR-D interface

  MAR-D is a mechanism in SAPHIRE to import or export PRA data. For example, models created in a different PRA engine can be placed in MAR-D format and loaded into SAPHIRE.

- User settings

  SAPHIRE has a user settings option to change settings for general display, general analysis interface, significance determination process interface, the rules editor, and the graphical editor.

- User interfaces

    In the SAPHIRE main screen, users can create and edit models, as well as perform PRA functions. This main screen is referred to as the Standard Analysis (SA) user interface. The SA interface is where the nominal PRA model resides. In Version 8, user interfaces have also been designed for Significance Determination Process (SDP) Phase 2 analyses with the SPAR models (SDP interface), events and conditions assessment (ECA interface), and more complex types of general analysis (GA interface). The nominal PRA model is loaded into the selected workspace to perform an analysis, where it is compared to the conditional cases resulting from the analysis.

- Help documentation

    SAPHIRE is installed with a hypertext-based user help manual.

A summary of the key features found in SAPHIRE versions 7 and 8 are listed in Table 1.

Table 1. Features offered in recent SAPHIRE versions.

Item	Description of Feature	Version 7.x	Version 8.x
A	Cut Set Sequence Generation		
A.1	Rule-based Fault Tree Linking	X	X
A.2	Linking of Small Tree Events	X	X
A.3	Linking of Large Tree Events	X	X
A.4	Sequence Capacity	2 million	2 million
B	Cut Set Generation		
B.1	Fault Trees	X	X
B.2	Event Trees	X	X
C	Cut Set Gathering		
C.1	Sequence End States	X	X
C.2	Sequence End State Cut Sets	X	X
D	Cut Set Partitioning		
D.1	End State Definition by rules	X	X
E	Cut Set Slice		
E.1	By Event	X	X
E.2	By Probability	X	X
E.3	By Rules	X	X
E.4	Multiple sequential slices		X
E.5	Enhanced cut set slice viewer		X
F	Cut Set Post-processing (Recovery)		
F.1	Event Trees	X	X
F.2	Fault Trees	X	X
F.3	Ability to layer rule application		X
G	Change Sets		
	(Selected subset of Basic Events for temporary analysis)		
G.1	Single event selection	X	X
G.2	Multiple event selection	X	X
G.3	Group event selection	X	X
H	Flag Sets		
	(Selected subset of Basic Events with logic changes only)		
H.1	Cut Set with Static Flag Sets	X	X
H.2	Cut Set with Dynamic Flag Sets (linkage rules)	X	X
H.3	Applicable to Fault Trees	X	X
H.4	Applicable to Sequences	X	X
H.5	Applicable to Fault Trees within Sequences	X	X
H.6	Flag sets can affect common-cause events		X
I	Cut Set Quantification Methods		
I.1	Minimal cut set upper-bound	X	X
I.2	Min-Max	X	X
I.3	Rare Event	X	X

Item	Description of Feature	Version 7.x	Version 8.x
I.4	Split Fraction (sequences only)	X	X
I.5	Binary Decision Diagram (fault trees)		X
J	**Cut Set Analysis**		
J.1	Cut Set Verification – cut sets solved, gathered, with truncation by size or probability, auto recovery	X	X
J.2	Cut Set path tracing	X	X
J.3	Cut Set comparison	X	
J.4	Fault Tree	X	X
J.5	Event Trees / Sequences	X	X
J.6	End States	X	X
K	**Basic Event Management**		
K.1	Basic Events – Generation	X	X
K.2	Basic Event – Templates	X	X
K.3	Multiple basic event editing at the same time		X
L	**Basic-Event Calculations**		
L.1	Compound Events Common-cause plug-in modules		
L.1.1	Common-cause alpha-factor module	X	X
L.1.2	Common-cause beta-factor module	X	X
L.1.3	Common-cause capacity load module	X	X
L.1.4	Common-cause multiple Greek letter module	X	X
L.1.5	Common-cause multiple group module	X	X
L.1.6	Common-cause alpha-staggered module	X	X
L.1.7	Common-cause RASP expanded module		X
L.1.8	Loss-of-offsite power module	X	X
L.1.9	Time series module	X	X
L.1.10	General calculation module	X	X
L.2	Failure Probability on Demand	X	X
L.3	Failure Probability to Run	X	X
L.4	Value input (for any value)	X	X
L.5	Failure Probability to Run w/ repair	X	X
L.6	Failure Probability to Run	X	X
L.7	House Event True (Prob = 1.0), i.e. failed	X	X
L.8	House Event False (Prob = 0.0), i.e. success	X	X
L.9	House Event Ignore	X	X
L.10	Compound Event	X	X
L.11	Human Factor Event	X	X
L.12	Fault tree Min Cut Upper Bound Value	X	X
L.13	End State Min Cut Upper Bound Value	X	X
L.14	Ground Acceleration Value	X	X
L.15	Hazard Curve	X	X
M	**Importance Measures**		
M.1	Fussell-Vesely Importance Measure	X	X

Item	Description of Feature	Version 7.x	Version 8.x
M.2	Birnbaum Importance Measure	X	X
M.3	Risk increase ratio importance measure	X	X
M.4	Risk reduction ratio importance measure	X	X
M.5	Risk increase interval importance measure	X	X
M.6	Risk reduction interval importance measure	X	X
M.7	Group importance measure	X	X
M.8	Uncertainty determination on Importance Measures	X	X
N	Model Creation		
N.1	Fire and flooding capability	X	X
N.2	Fault Tree text editor	X	
N.3	Drag-and-drop Fault Tree graphical editor		X
N.4	Event Tree text editor	X	
N.5	Event Tree graphical editor	X	X
N.6	Drag-and-drop Event Tree editor		X
N.7	Basic Load / Extract Data Models	X	X
N.7.1	Extract All	X	X
N.7.2	Load All	X	X
N.7.3	Extract All File types	X	X
N.7.4	Load All / Group	X	X
N.7.5	Fault Tree Logic	X	X
N.7.6	Designate output folder location	X	X
N.8	Graphical Export to Windows metafiles	X	X
N.9	Graphical Export to JPEG		X
N.11	Database Recovery	X	X
N.12	Database MAR-D Load and Extract	X	X
N.12.1	Event Tree MAR-D	X	X
N.12.2	Fault Tree MAR-D	X	X
N.12.3	Basic Event MAR-D	X	X
N.13	Macro manager		X
N.14	Alternate names and descriptions for all database objects (for multilingual use)	X	X
N.15	Model Version Upgrade (backward compatible)	X	X
N.16	Integrate two projects into single project		X
N.17	Creation of external events model via Accident Sequence Matrix file		X
N.18	Phase-aware basic events and event trees		X
N.19	User definable grouping of event trees		X
O	Model Creation Logic Gate Types		
	(Maximum inputs 256 unless otherwise specified)		
O.1	AND	X	X
O.2	OR	X	X
O.3	N of M (Max N=98 , Max M=99)	X	X
O.4	NAND (Not AND)	X	X

Item	Description of Feature	Version 7.x	Version 8.x
O.5	NOR (Not OR)	X	X
O.6	Transfer Gate	X	X
O.7	Inhibit gate	X	
P	**Uncertainty Calculations** **(Monte Carlo and Latin Hyper Cube Sampling)**		
P.1	None (or Point Value only)	X	X
P.2	Normal Distribution	X	X
P.3	Lognormal Distribution	X	X
P.4	Beta Distribution	X	X
P.5	Chi Squared Distribution	X	X
P.6	Exponential Distribution	X	X
P.7	Uniform Distribution	X	X
P.8	Constrained non-informative Distribution	X	X
P.9	Gamma Distribution	X	X
P.10	Maximum Entropy Distribution	X	X
P.11	Dirichlet Distribution	X	X
P.12	Seismic Log Normal analysis	X	X
P.13	Histogram Distribution	X	X
P.14	Triangular Distribution	X	X
Q	**Uncertainty Calculations (Parameter Settings)**		
Q.1	Seed	X	X
Q.2	Sample Size	X	X
Q.3	Built in cumulative and density plots		X
R	**General Support Features**		
R.1	Sensitivity Wizard	X	X
R.2	Importance Measures Wizard	X	X
R.3	Embedded Macro capability	X	X
R.4	Editing User Information	X	X
R.5	Page numbering control on graphic format	X	X
R.6	Conversion from alpha to graphic format	X	n/a
R.7	On-line Context Sensitive help	X	X
R.8	User customizable icons calling analysis icons		X
R.9	Multiple editing and reporting windows open at the same time		X
R.10	Project Check		X
R.11	Project-wide search ability		X
R.12	Bookmarking of object lists		X
R.13	Drag-and-drop flag and change set creation		X
R.14	Support for opening and creating compressed (zip) project files		X
S	**General Support Features (Report Generation)**		
S.1	Project Reports	X	X
S.2	Attributes	X	X
S.3	Basic Event	X	X

Item	Description of Feature	Version 7.x	Version 8.x
S.4	Fault Tree	X	X
S.5	Event Tree	X	X
S.6	End State	X	X
S.7	Sequence	X	X
S.8	Change Set	X	X
S.9	Flag Set	X	X
S.10	Gate	X	X
S.11	Histogram	X	X
S.12	Slice	X	X
S.13	User Info	X	X
S.14	Cross Reference Reports	X	X
T	Report Format Types		
T.1	ASCII	X	X
T.2	RTF	X	X
T.3	HTML	X	X
T.4	Acrobat™ PDF		X
U	General Analysis Types		
U.2	Initiating Event Analysis	X	X
U.3	Condition Assessment Analysis	X	X
U.4	Accident Sequence Precursor	X	X
U.4.2	General analysis types	X	X
U.4.3	Load-capacity calculation module	X	X
U.5	Significance Determination Process		X
V	Application Program Interface		
V.1	Microsoft Visual Basic™ and VBA interface	X	X
V.2	Microsoft Visual C\C++™ interface	X	X
V.3	Borland Delphi™	X	X
W	Operating Systems		
W.1	Windows XP™	X	X
W.2	Windows Vista™	X	X
W.3	Windows 7™	X	X

2. PROJECT ANALYSIS WITH THE BASE CASE MODEL

Solving a SAPHIRE Version 8 project can be performed directly in the project or by setting up a new analysis in a user selectable workspace. The differences between these two analysis options will be discussed below.

The starting point in analyzing an existing model is to load the project into SAPHIRE Version 8 as shown in Figure 1. The project specified in the "Current Project" field is opened or a new project can be selected by selecting the Browse option and scrolling through the folders on the hard drive until the project is located. The demonstration models included with the SAPHIRE software package are used in this report.

The next step to opening the project is to specify what type of analysis is going to be performed. To perform an analysis, select a user interface from the drop down box and then click the green arrow below this option. Alternatively, select the "Modify a Reliability or Risk Analysis Project" option. The "modify" option will present a user interface (the SA interface) which allows for project creation and manipulation along with all types of standard probabilistic risk analyses. In addition, it provides a link to the user interfaces given under the dropdown box mentioned previously.
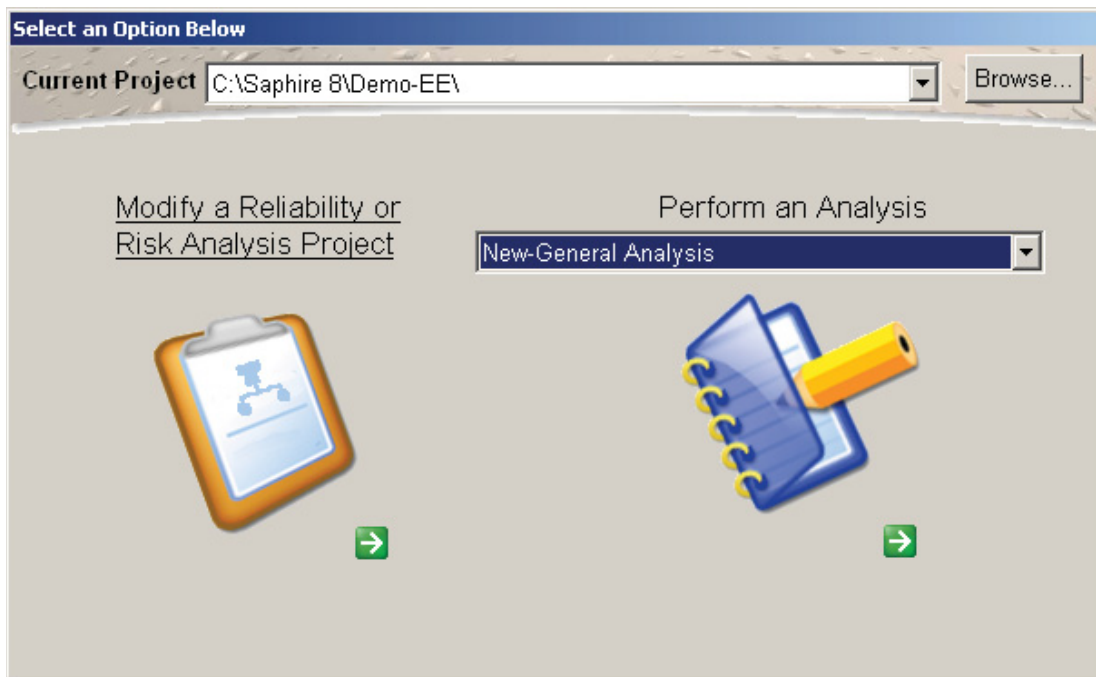

Figure 1. Starting dialog for SAPHIRE 8

2.1 Model Building and Analysis

SAPHIRE Version 8 has provided an improved interface to modify a reliability or risk analysis project's database. Figure 2 shows the SA user interface. The windows in the left panel of the

main screen are expandable and collapsible (which the user may customize by showing or hiding under the View option). The whole left panel can be resized as well by clicking and dragging the edge. Windows in the left panel can be used for model manipulation and for creating and running analyses.
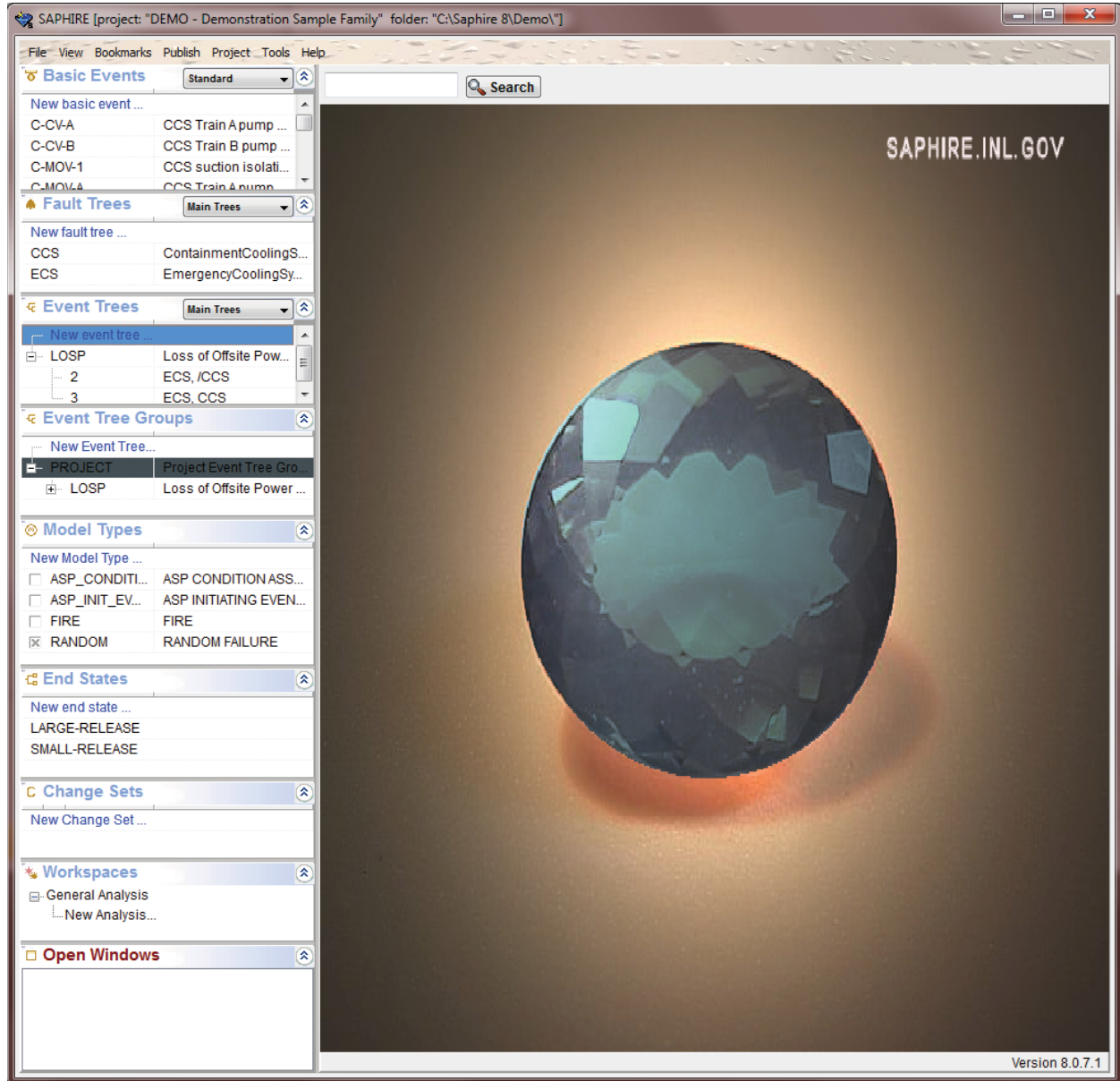

Figure 2. SAPHIRE 8 screen showing all left panel views

Note that in Figure 2, the SAPHIRE version number is shown in the lower right corner of the main window. If an Internet connection is available, SAPHIRE will check the SAPHIRE web site for the most up-to-date released version. If these two versions are different, this version information will be displayed in the lower right corner. The user can view the portions of the

probabilistic risk assessment model on the left hand side by selecting "View" on the toolbar. The view options are shown in Figure 3.
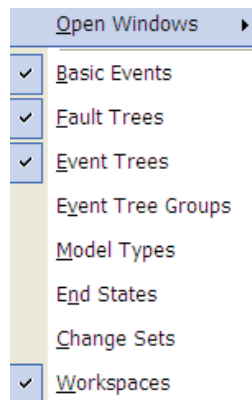

Figure 3. View options

2.1.1 Basic Event Editing

The basic event editor is segregated into five tabs for the different types of basic event information.

- Failure Model
- Attributes
- Applicability
- Notes
- Summary

Failure Model Tab

The "Failure Model" option allows the user to edit the failure model(s) for basic events. The "Failure Model" page consists of a tree-view control listing all of the applicable options required to specify the event's failure model. The Calculated Probability is determined from the settings below it.

In SAPHIRE Version 8, a failure model is assigned to each event's Model Type (if more than one, as assigned in the "Applicability" tab) as shown in Figure 4. Each failure model can have a description.
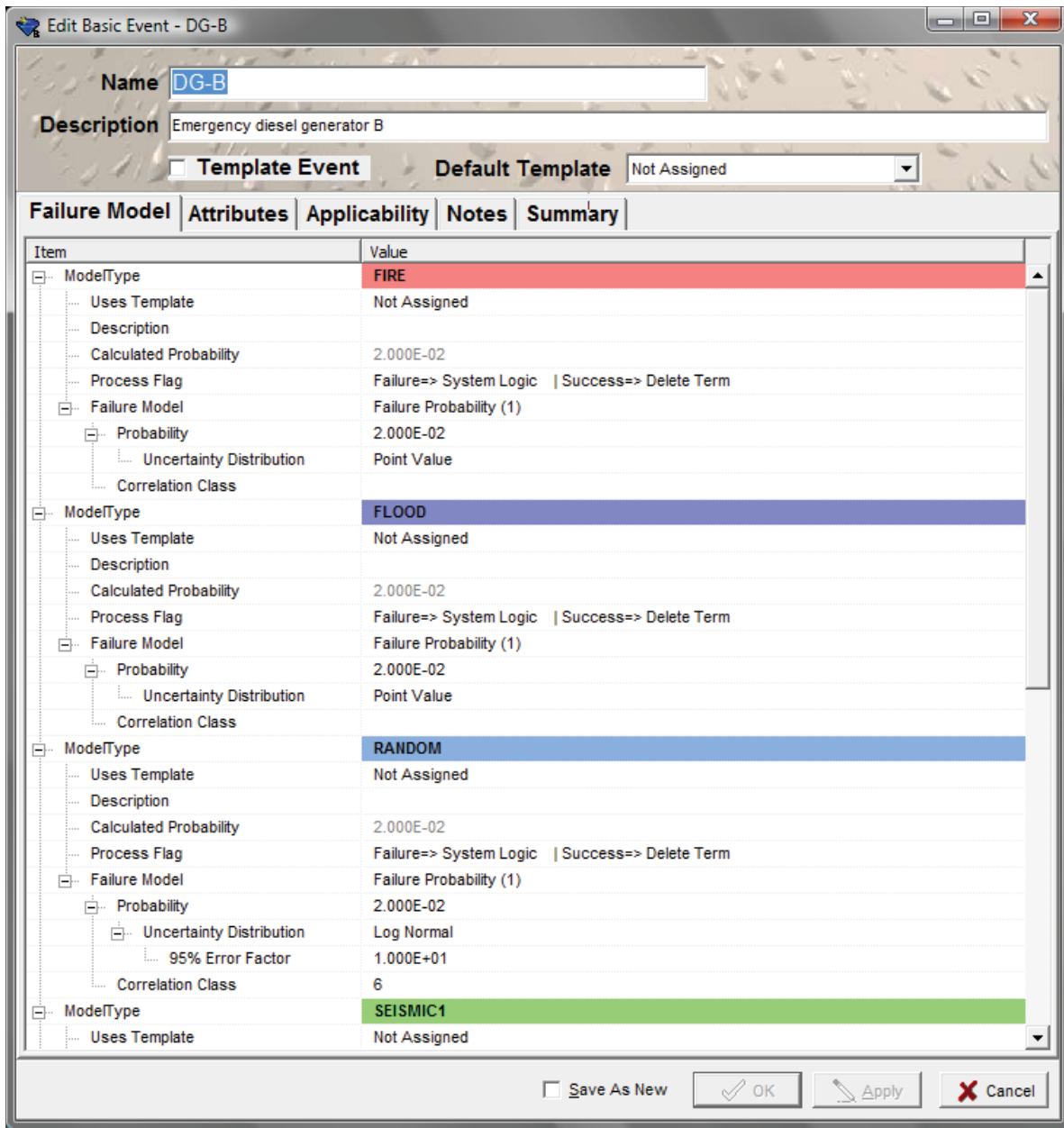
Figure 4. Basic event editor screen showing multiple "virtual" basic events

Attributes Tab

In this tab the basic event can be specified as a Template Event. The following can also be set:

- Event Type
- General Event
- Hazard Event
- Initiating Event

- Recovery Event

Note: In Version 8 the Initiating Event should be set to type "N," Initiating Event Frequency, under the Failure Model Tab -> Failure Model dropdown box. This calculation type allows the initiating event frequency units to be explicitly specified, for example in terms of "per year" or "per mission."

Applicability Tab

The applicability tab controls the basic events "Model Type" and "Phase." Check the box next to each applicable model type or phase.

Notes Tab

The model developer can specify notes such as the references for the BE data.

Summary Tab

This tab provides a summary of the basic event information. Included in the summary report is a graphical depiction of the uncertainty.

Virtual Basic Events

As noted above each basic event failure mode will be displayed in the Failure Model Tab along with the parameter data for each model type. This graphical display is a Virtual Basic Event. This is because while the data for all model types belonging to the basic event are displayed in the Failure Model Tab, Version 8 carries the different model types along in the fault tree as separate basic events distinguished by a suffix (e.g., A-FIR for fire, A-EQ for seismic, etc). Virtual basic events allow different reliability models and other parameters corresponding to different initiating events to be viewed in one location, improving the ability to review complex PRA models.

Automatically Generated Basic Events

SAPHIRE 8 will automatically generate basic events to support a new common cause failure probability method. These basic events can be viewed from a drop down box on the main screen.

2.1.2 Fault Tree Editing

The capability to modify the model and the functionality of the graphical editor has improved. The list of Fault Trees is shown in the fault tree window in the left panel. To open a Fault Tree, double click on the name. As shown in Figure 5 for the Demo CCS Fault Tree, the basic events can be stacked. Transfers can also be stacked. The transfer icon is right-clicked to follow the transfer by opening up the transfer tree graphics. SAPHIRE Version 8 keeps a list of all of all the open windows. Multiple objects are kept accessible as all open trees and other windows are listed in the Open Windows window at the bottom of the left panel. Basic Events can be modified through choosing them in the Basic Events window or within the graphical editor.
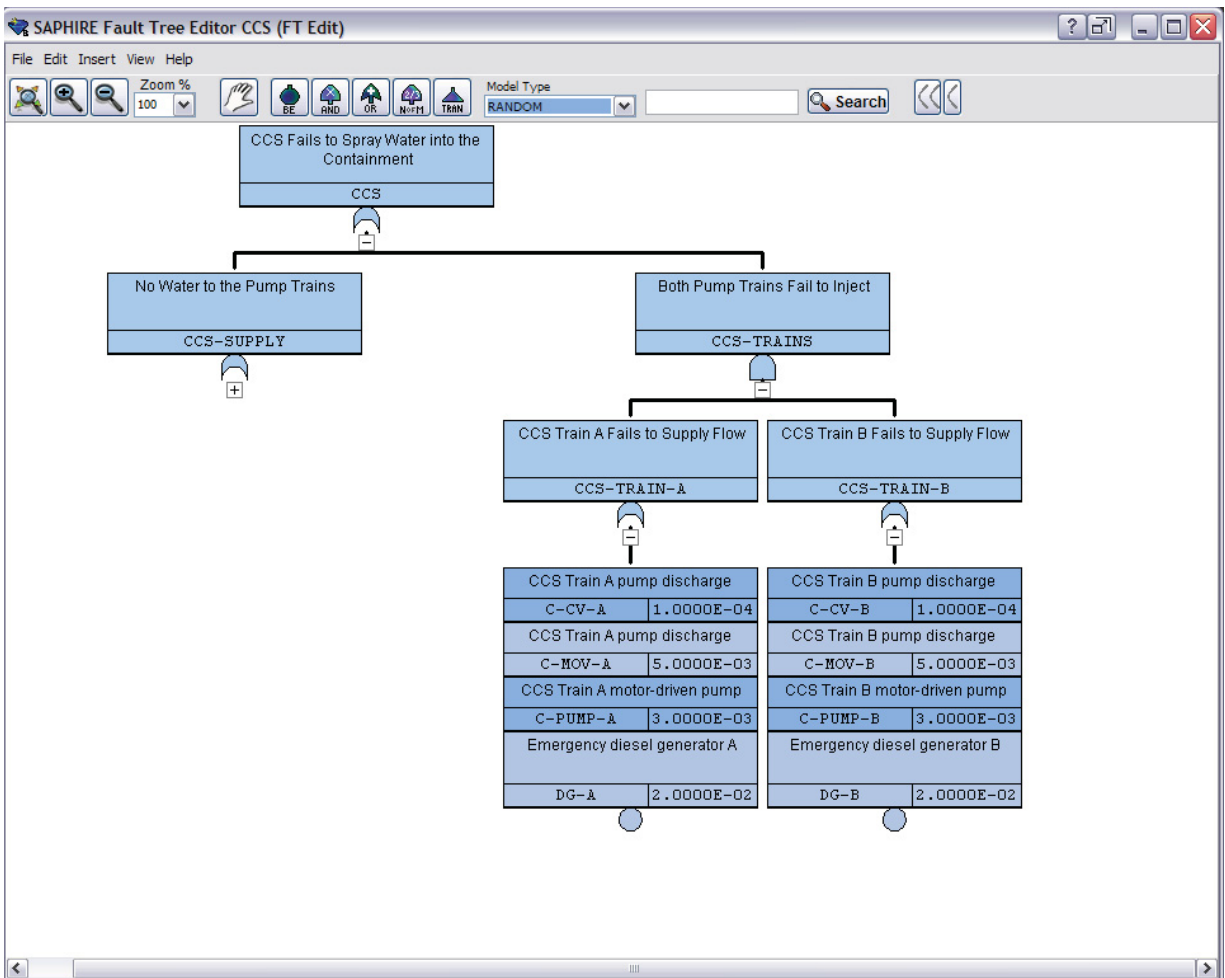

Figure 5. Fault Tree graphical editor in SAPHIRE 8

Portions of a Fault Tree are also collapsible by clicking on the "-" icon underneath a gate. A "+" icon indicates there are unexpanded gates and/or events, which then are clicked to expand the model. Adding a gate to a Fault Tree is accomplished by clicking on the button with the desired

gate icon at the top of the editing screen. The cursor will show an arrow with a null symbol until it is placed over a gate that can accept the insertion. The cursor will then show a gate symbol – a left mouse click inserts the gate (see Figure 6).
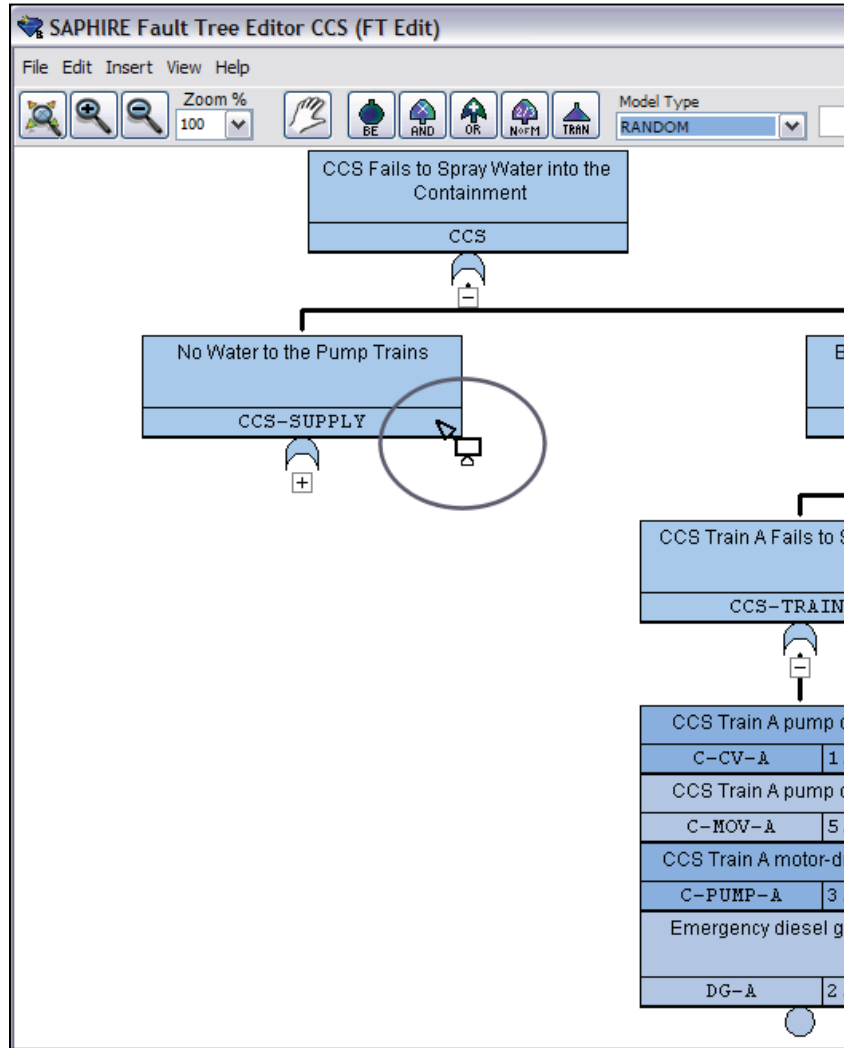

Figure 6. AND Gate cursor in Fault Tree editing screen

2.1.3 Event Tree Editing

Event Tree editing is accomplished in much the same manner as Fault Tree editing. All of the increased functionality of multiple object accessibility is available as described in the Fault Tree Editing section. Event Trees and Fault Trees can be open at the same time. The branch and top event insertions are made by clicking on the icon for the chosen insertion and placing the resulting cursor (branch up, branch down, top event insert left, top event insert right) over the branch point or top event as applicable, then click to insert. Information and editing of a top

event can be accessed by double clicking on the top event box as in SAPHIRE Version 7. When end-states are double clicked, the frequency and sequence name is also viewable (see Figure 7).
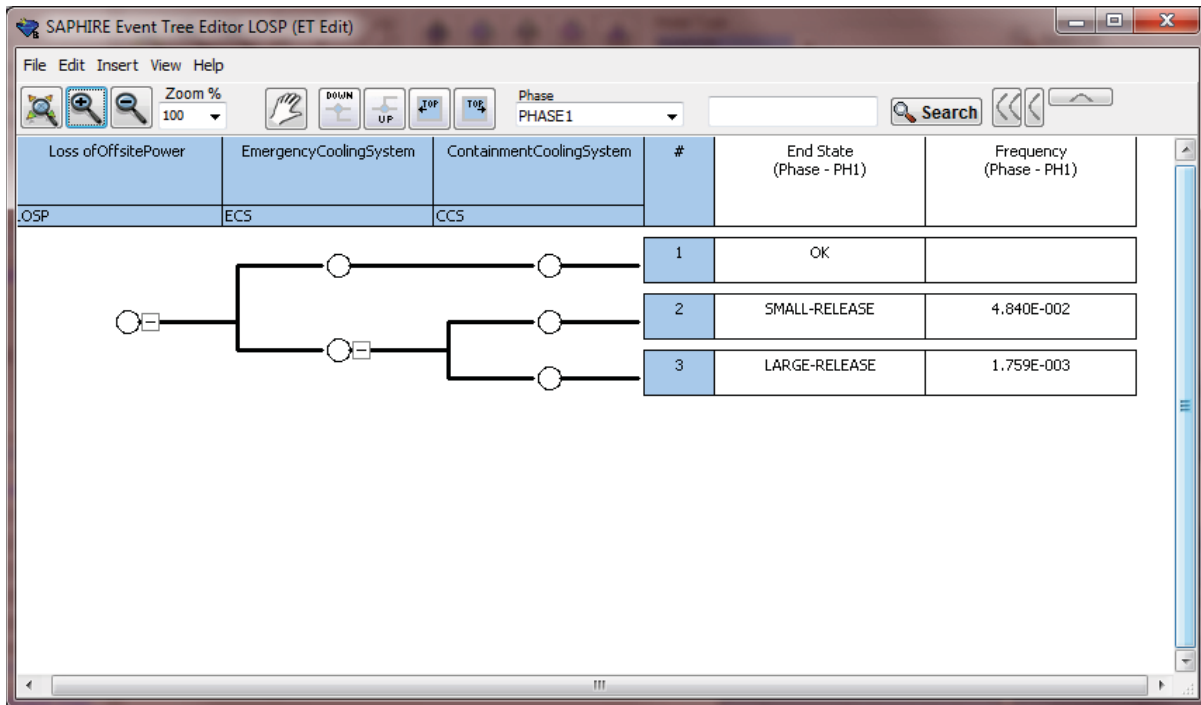

Figure 7. Event Tree editing screen

2.2   Fault Trees

2.2.1 Generate Cut Sets

To solve fault tree cut sets from an existing model, the fault tree(s) need to be highlighted and then right click the mouse.   A pop-up menu will appear. Select **Solve**.

The Solve option is different compared to SAPHIRE Version 7 primarily in that SAPHIRE Version 8 has all of the solve options located on this one option screen (see Figure 8). Within this screen the following applications are performed: cut set truncation, flag settings, quantification method to be used, number of computer "threads" (processors) to be used concurrently, starting gate options, cut set generation, apply recovery rules, cut set update, cut set quantification, clearing the current case, and nominal-case update. Also, the different model types can be analyzed together or separately.
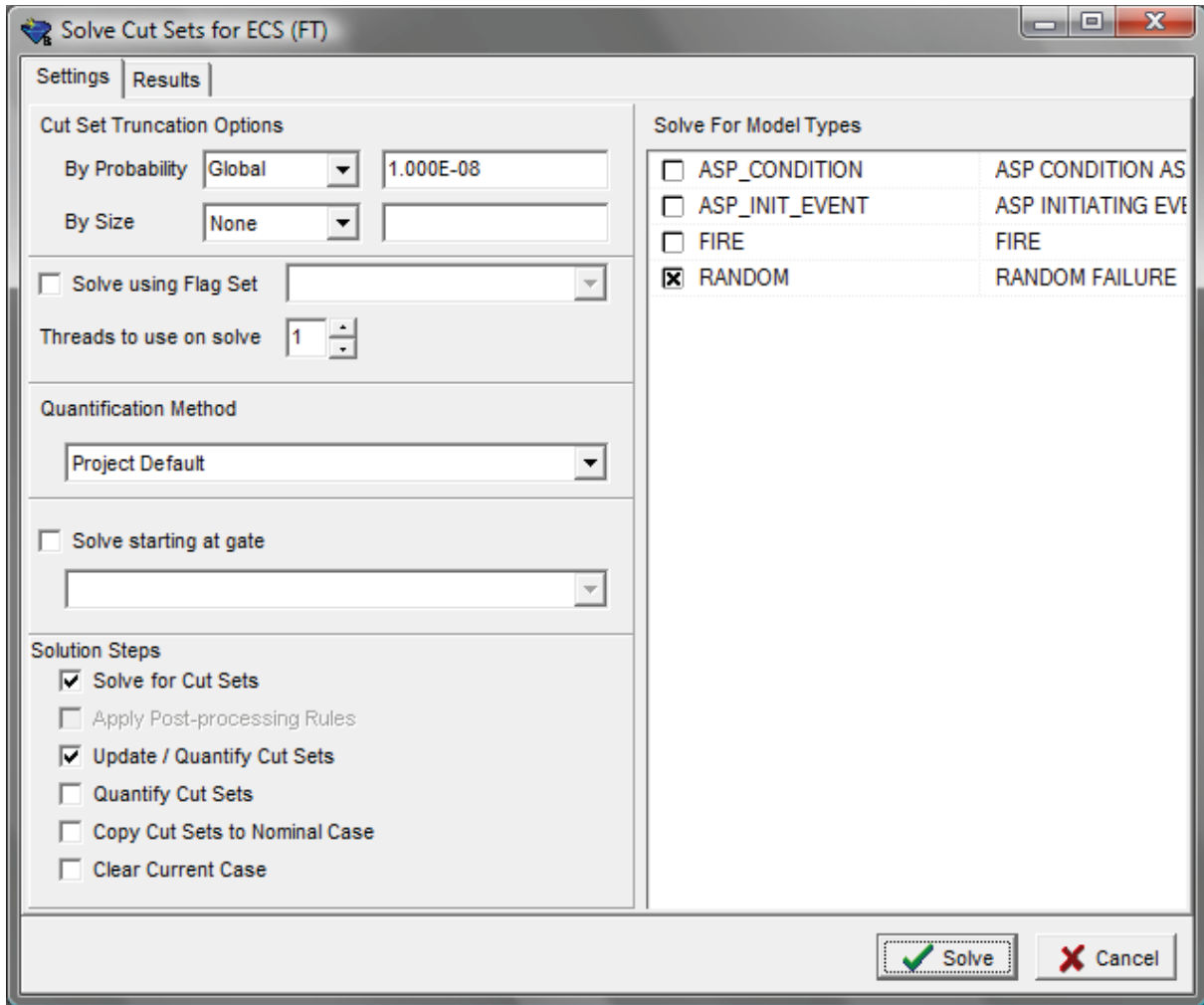
Figure 8. The fault tree solve settings

Note that in SAPHIRE Version 8, the different model types (e.g., random, seismic, fire, flood) can be solved at the same time (as opposed to SAPHIRE Version 7). Whenever more than one model type is solved, the cut sets for that particular group of model types is stored uniquely from other groups. For example, if the random and flood model types are selected and solved, cut sets applicable to those two model types will be solved and stored internally in SAPHIRE Version 8 as a random+flood group. If random and fire model types are selected and solved, these new cut sets will be stored in a random+fire group, leaving the random+flood cut sets untouched for later retrieval.

2.2.2 Solve Options

Cut Set Generation

SAPHIRE solves the selected fault tree(s) based on its logic to obtain the minimal cut sets. These resultant cut sets are then quantified or truncated based on the truncation options. In

general, the cut sets are stored in the "current case" by default (although the user may specify that an additional copy of the cut sets be stored in the "nominal case").

Cut Set Truncation

The truncation options are the same as those from SAPHIRE Version 7. The cut sets can be solved at a Global truncation or a System-Specific truncation. The Global truncation will solve all fault trees at the specified truncation (i.e., globally). The System-Specific truncation will solve the fault trees at the truncation level saved for each fault tree. This second option requires that a specific truncation be stored for all fault trees. The last truncation option is NONE, which means no truncation will be used in the cut set generation.

The next set of truncation options deal with the number of basic events per cut set. This truncation option is Size (direct number of basic events from logic model).

Solution Steps

This set of check boxes lets SAPHIRE Version 8 know what type of solution is requested. Whichever, box(es) is checked, that particular operation will be performed. Multiple boxes can be selected or just one box depending upon the final result. Each box performs the following process:

Solve for Cut Sets – Solve the fault tree for minimal cut sets based on the fault tree logic model.

Apply Post-Processing Rules – This check box applies all applicable recovery rules to the fault tree cut sets (Fault Tree specific rule first then Project rule). Note that the option may be disabled if no post-processing rules exist for the fault tree(s).

Update/Quantify Cut Sets – This option re-evaluates the existing cuts to remove any potential non-minimal cut sets and then quantifies these cut sets using the default quantification method.

Copy Cut Sets to Nominal Case – This option stores the resultant cut sets into the nominal case.

Clear Current Case – This option deletes any stored cut sets in the current case.

Model Types

The last option within this screen is what Model Types are to be solved. Each basic event/fault tree can be susceptible to one or more of the Model Types available within SAPHIRE. The default is random, but multiple Model Types can be selected and SAPHIRE will solve the fault tree based on the Model Type(s) selected. The results will all be grouped together based on the Model Type(s) selected.

2.2.3 Quantification

The same quantification options are available in SAPHIRE Version 8 as in SAPHIRE Version 7. These quantification options are Minimal Cut Set Upper-bound, Rare Event, and Min-Max. One way to change these options is by right clicking the mouse on the selected fault tree (only one may be selected) and selecting 'Edit Logic.' Inside the Fault Tree Editor, select 'Edit' from the menu bar, and then 'Properties'. The default quantification method may be changed in the lower left corner of the model information editor (as shown in Figure 9).
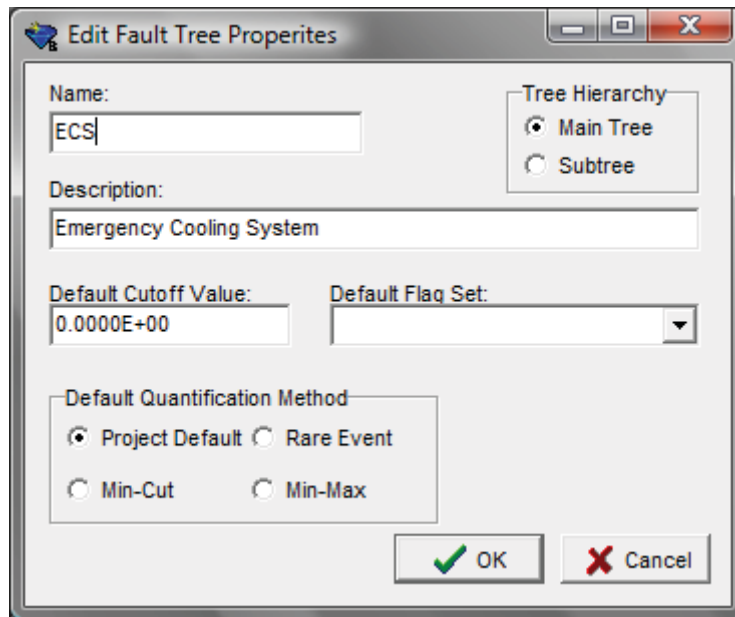

Figure 9. Default quantification selection

Report

There are multiple Report options (available by right-clicking on an object):

- View Summary Results
- View Cut Sets
- View Importance Measures
- View Uncertainty

The other option is to select Publish from the project's main menu bar (for example, the Publish option for fault trees is shown in Figure 10 and 11). This report function allows the user to specify from a default listing of report outputs and select the document structure for printing. Note that the following output formats are supported for most types of reports (if a format is not applicable to a specific type of report, it will be disabled):

- Hyper-Text Markup Language (.htm)
- Adobe Acrobat (.pdf)
- Rich Text Format (.rtf)
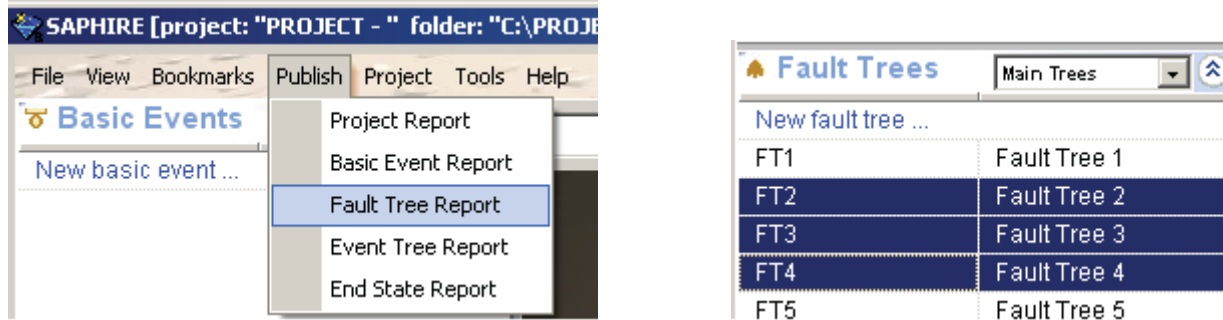- ASCII-based comma delimited (.txt)
- Excel (.xls)
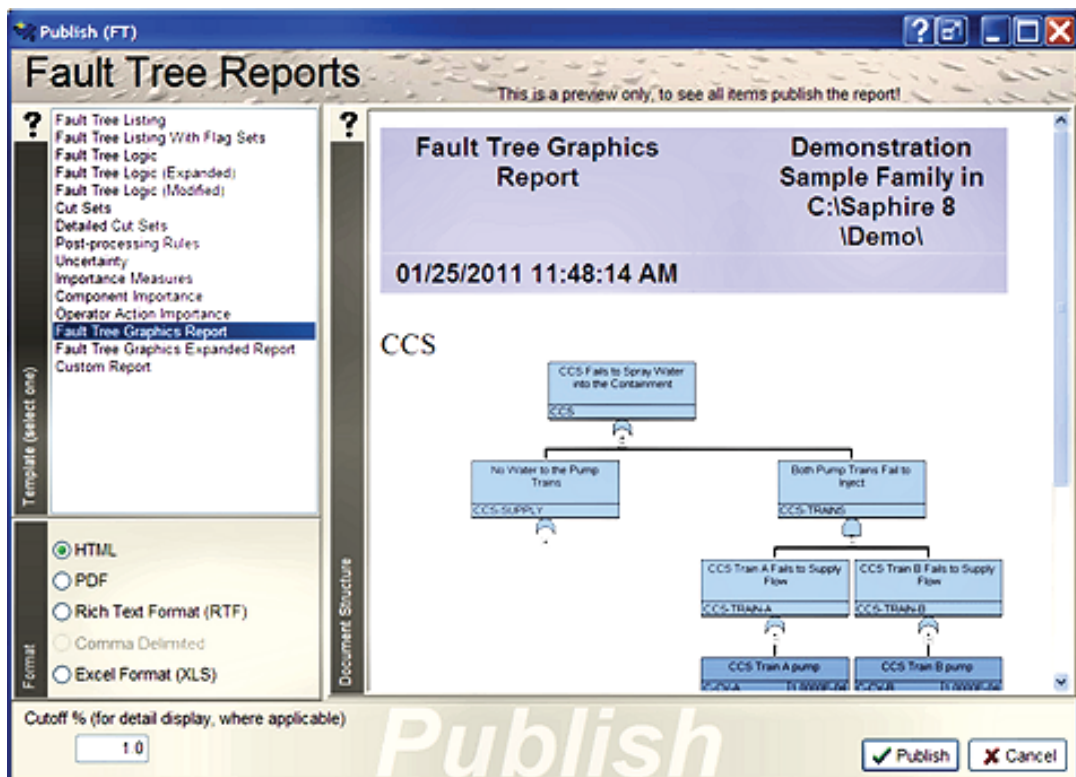

Figure 10. Fault tree report path

Figure 11. Fault tree reports (graphic report selected)

Uncertainty Analysis

The uncertainty analysis is primarily the same as that performed in SAPHIRE Version 7. The only differences are the menu options and an improved plot. In SAPHIRE Version 8, the uncertainty analysis is performed in the **View Uncertainty** option after the cut sets have been generated.

2.2.4 Importance Measures

The importance measures in SAPHIRE Version 8 are the same as in SAPHIRE Version 7. The only difference is the menu option and the type of outputs. However, two improvements to importance measures in the current SAPHIRE Version 7 are carried into Version 8 as discussed below.

First, the exact expression for the Fussell-Vesely (FV) is used versus an approximation. The FV measure is an indication of the percentage of the minimal cut set upper bound contributed by the cut sets containing the basic event. The equation for FV importance now being used (in the current SAPHIRE Version 7 and Version 8) is

$FV = F(i) / F(x)$

where

F(i) is the minimal cut set upper bound for the group of cut sets containing the event

F(x) is the original minimal cut set upper bound

In older versions of SAPHIRE (prior to the current Version 7), the FV equation used an approximate expression, $FV = 1 - F(0)/F(x)$, where

F(0) is the minimal cut set upper bound with the event probability set equal to 0.0.

F(x) is the original minimal cut set upper bound.

The difference between the FV exact equation and the FV approximation can be seen for basic events with large failure probabilities.

Second, uncertainty analysis can be performed on importance measures. The importance measures can be reported as point estimates or the component variability can be incorporated into the results. The point estimate is calculated using the formal importance measure equations using just the input probabilities for the individual components and the overall point estimate for the fault tree. However, if the importance of the components based on their variability is required, SAPHIRE Version 8 will perform the same calculation as stated for the point estimate except now it samples from the component's uncertainty and feeds this into the importance measure equation. For example, the standard importance measures are calculated as

Fussell-Vesely: $FV = F(i) / F(x)$

Risk Increase Ratio: $RIR = F(1) / F(x)$

Risk Reduction Ratio: $RRR = F(x) / F(0)$

Birnbaum: $B = F(1) - F(0)$

where:

$F(x)$ = value of all the minimal cut sets evaluated with the basic event probabilities at their mean value.

$F(i)$ = value of the minimal cut sets that contain the i'th basic event.

$F(0)$ = value of all the minimal cut sets evaluated with the i'th basic event set to zero.

$F(1)$ = value of all the minimal cut sets evaluated with the i'th basic event probability set to one.

For the point estimate, the equation is solved only once for each component using their input probability. Now to take into consideration the component's variability, the importance measure equation is calculated for each sample. From the results of the samples, the importance measure mean, and other parameters are obtained for each basic event.

The report for the importance measure based on the point estimate is shown in Figure 12.
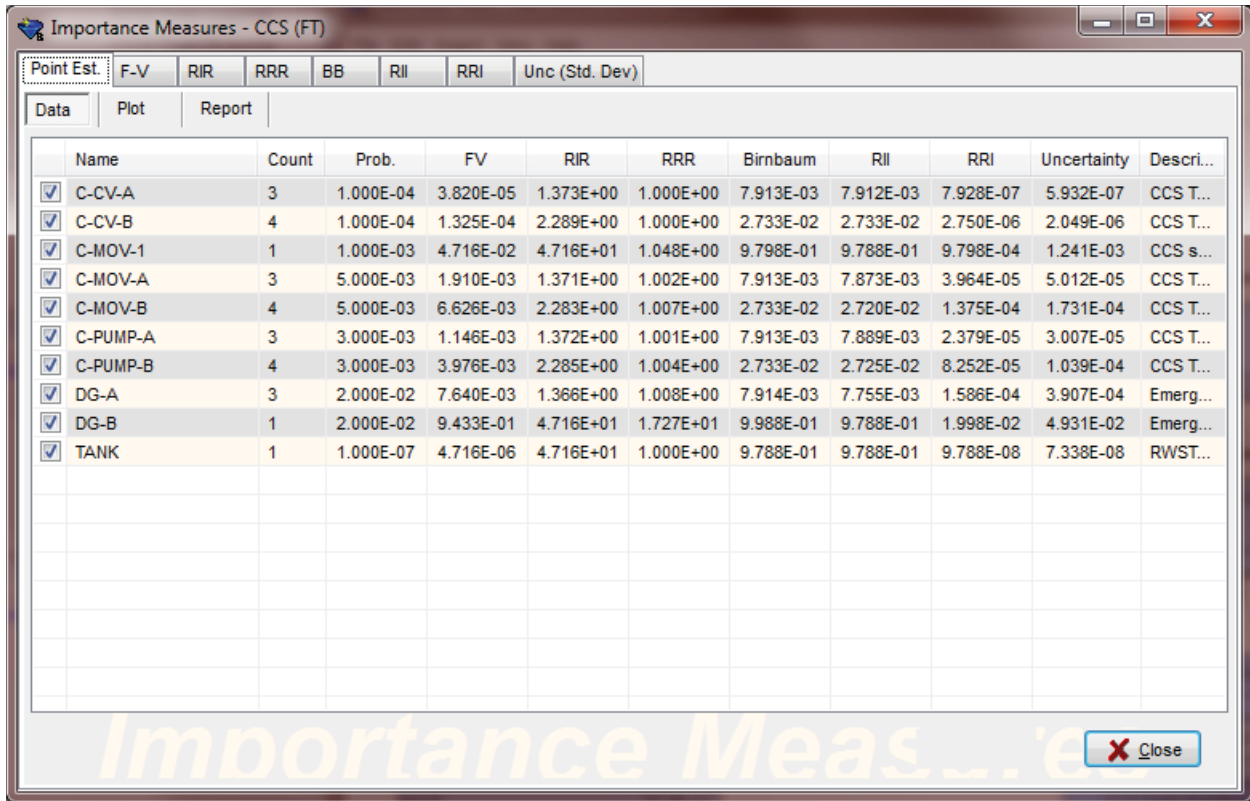
Name	Count	Prob.	FV	RIR	RRR	Birnbaum	RII	RRI	Uncertainty	Descri...
☑ C-CV-A	3	1.000E-04	3.820E-05	1.373E+00	1.000E+00	7.913E-03	7.912E-03	7.928E-07	5.932E-07	CCS T...
☑ C-CV-B	4	1.000E-04	1.325E-04	2.289E+00	1.000E+00	2.733E-02	2.733E-02	2.750E-06	2.049E-06	CCS T...
☑ C-MOV-1	1	1.000E-03	4.716E-02	4.716E+01	1.048E+00	9.798E-01	9.788E-01	9.798E-04	1.241E-03	CCS s...
☑ C-MOV-A	3	5.000E-03	1.910E-03	1.371E+00	1.002E+00	7.913E-03	7.873E-03	3.964E-05	5.012E-05	CCS T...
☑ C-MOV-B	4	5.000E-03	6.626E-03	2.283E+00	1.007E+00	2.733E-02	2.720E-02	1.375E-04	1.731E-04	CCS T...
☑ C-PUMP-A	3	3.000E-03	1.146E-03	1.372E+00	1.001E+00	7.913E-03	7.889E-03	2.379E-05	3.007E-05	CCS T...
☑ C-PUMP-B	4	3.000E-03	3.976E-03	2.285E+00	1.004E+00	2.733E-02	2.725E-02	8.252E-05	1.039E-04	CCS T...
☑ DG-A	3	2.000E-02	7.640E-03	1.366E+00	1.008E+00	7.914E-03	7.755E-03	1.586E-04	3.907E-04	Emerg...
☑ DG-B	1	2.000E-02	9.433E-01	4.716E+01	1.727E+01	9.988E-01	9.788E-01	1.998E-02	4.931E-02	Emerg...
☑ TANK	1	1.000E-07	4.716E-06	4.716E+01	1.000E+00	9.788E-01	9.788E-01	9.788E-08	7.338E-08	RWST...

Figure 12. Point estimate importance measure dialog

The report for the importance measures allowing the component's uncertainty to be propagated is shown in Figure 13, where the report provides the results of the FV sampling. For each iteration (up to the number of samples specified) SAPHIRE determines the F(1), F(i), F(0), and F(x) terms previously defined, wherein each of the basic events in the cut sets are sampled using their individual epistemic distributions (e.g., lognormal, beta, gamma). Then, for each iteration, the importance measures are quantified using these "F( )" terms. Each value for the various importance measures is stored in an array that is later used to determine the moments (such as mean and standard deviation) and percentiles (such as 5th, 50th, and 95th).
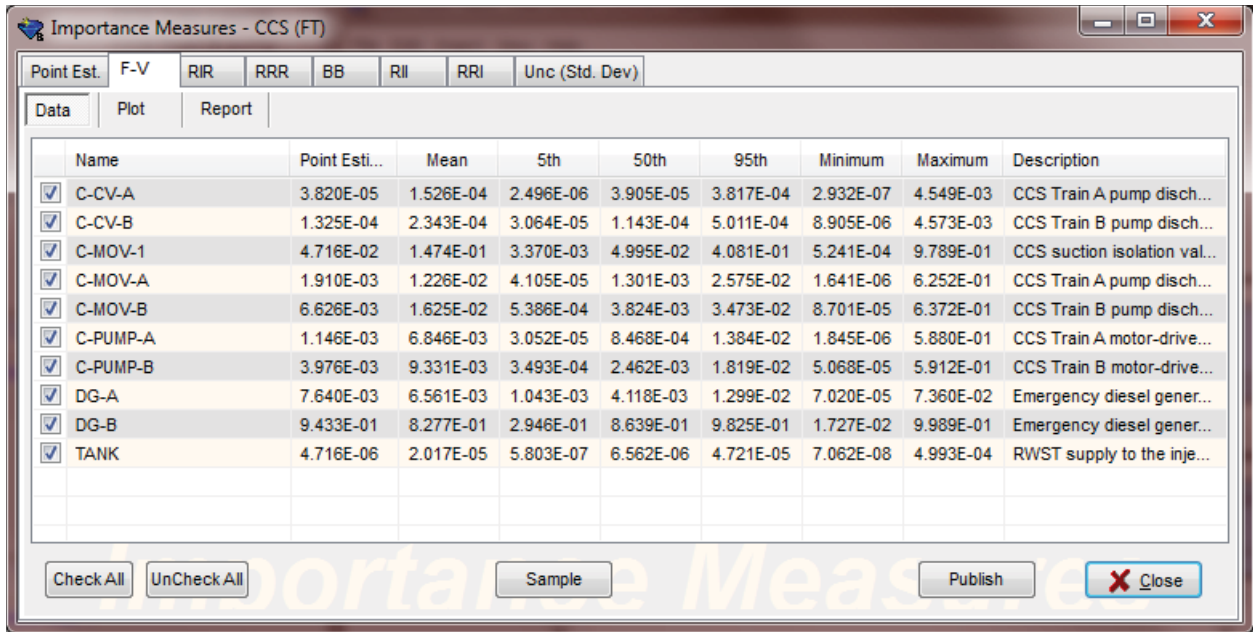
29

Figure 13. Uncertainty on importance measures

The calculation arrays are temporarily (due to the potentially large storage space) stored in memory – consequently, once the user closes the importance measure screen, the uncertainty results are discarded and must be rerun if needed again.

The other report option for the importance measures is a plot of each component based on the sampling process. This option is displayed by selecting the plot tab (see Figure 14).
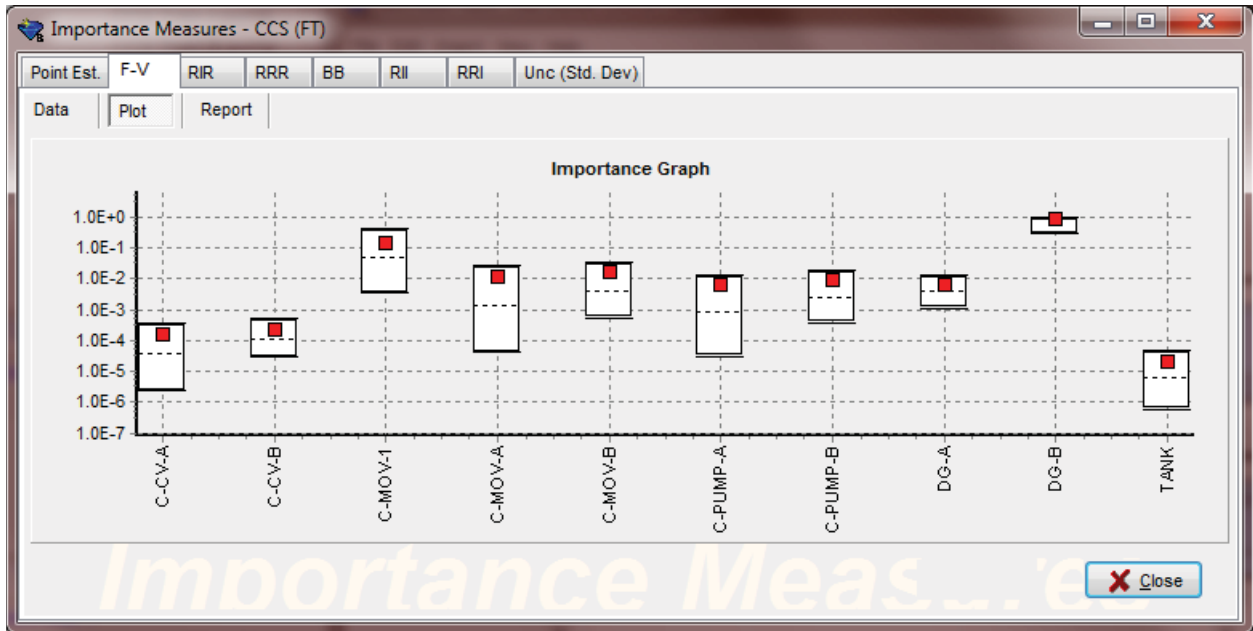

Figure 14. Plotting of importance measure uncertainty

2.3   Event Trees

The analysis process for event trees is designed to solve the accident sequences for minimal cut sets. The cut set generation and other processes are similar to the fault tree option with exception that event tree sequences are used as the logic model. The different aspects of analyzing the event tree accident sequences will be covered.

2.3.1 Solve Options

Cut Set Generation

The cut set generation is based on the linked fault tree methodology (i.e. fault tree logic linked to the event tree logic). The solve option, available by right-clicking on an event tree, (shown in Figure 15) will generate the minimal cut sets for the accident sequence logic, which is then quantified. Depending if the event tree is highlighted or an individual sequence will determine which sequences are analyzed.
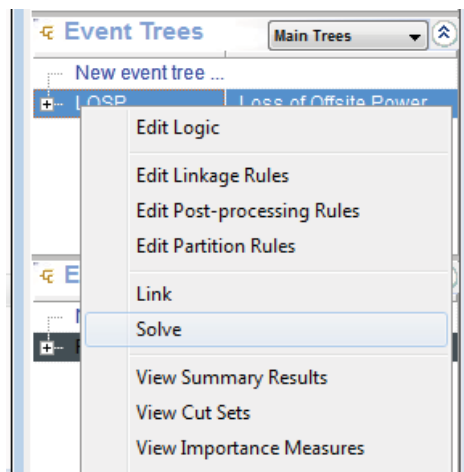
Figure 15. Event Tree Cut Set solve option path

Once the **Solve** option has been selected, SAPHIRE will pop up a screen that allows for sequence analysis settings. Each of the options will be discussed.

Cut Set Probability Truncation

      None –   This truncation option solves for all possible cut sets.

      Normal –   This truncation option generates all cut sets above the truncation value specified.

      Conditional – This truncation option generates all cut sets above the truncation value specified, ignoring the initiating event frequency.

Solution Steps

The solution steps check boxes are designed to specify how to solve the accident sequences. These check boxes can be checked individually and the Solve button clicked or multiple boxes can be checked and SAPHIRE will apply all operations that are checked starting at top and working down the list.

Solve for Cut Sets – Solve the event tree sequences that are highlighted (or event trees) for minimal cut sets based on the sequence logic model.

Apply Post-Processing Rules – This check box will apply all post-processing rules to the newly generated cut sets (Sequence specific rule, Event Tree specific rule, then Project rule). Note that the option may be disabled if no post-processing rules exist for the sequence(s).

Update/Quantify Cut Sets – This option re-evaluates the existing cuts to remove any potential non-minimal cut sets and then quantifies these cut sets using the default quantification method.

Quantify Cut Sets – This option quantifies the cut sets using the default quantification method.

Clear Cut Set Partitions – This option resets the partition rules (i.e., removes the end states from the cut sets that was created using partition rules).

Apply Partition Rules – This option applies the partition rules to create user specific end states for the cut sets based on rules (Sequence specific rule, Event Tree specific rule, then Project rule). Note that the option will be disabled if no post-processing rules exist for the sequence(s).

Copy Cut Sets to Nominal Case – This option stores the resultant cut sets into the nominal case.

Clear Current Case – This option deletes any stored cut sets in the current case.

Model Types

The last option within this screen is what Model Types are to be solved. Each basic event/fault tree/event tree can be susceptible to one or more of the Model Types available within SAPHIRE. The default is Random, but multiple Model Types can be selected and SAPHIRE will solve the fault tree based on the Model Type(s) selected (see Figure 16). The results will all be grouped together based on the Model Type(s) selected.
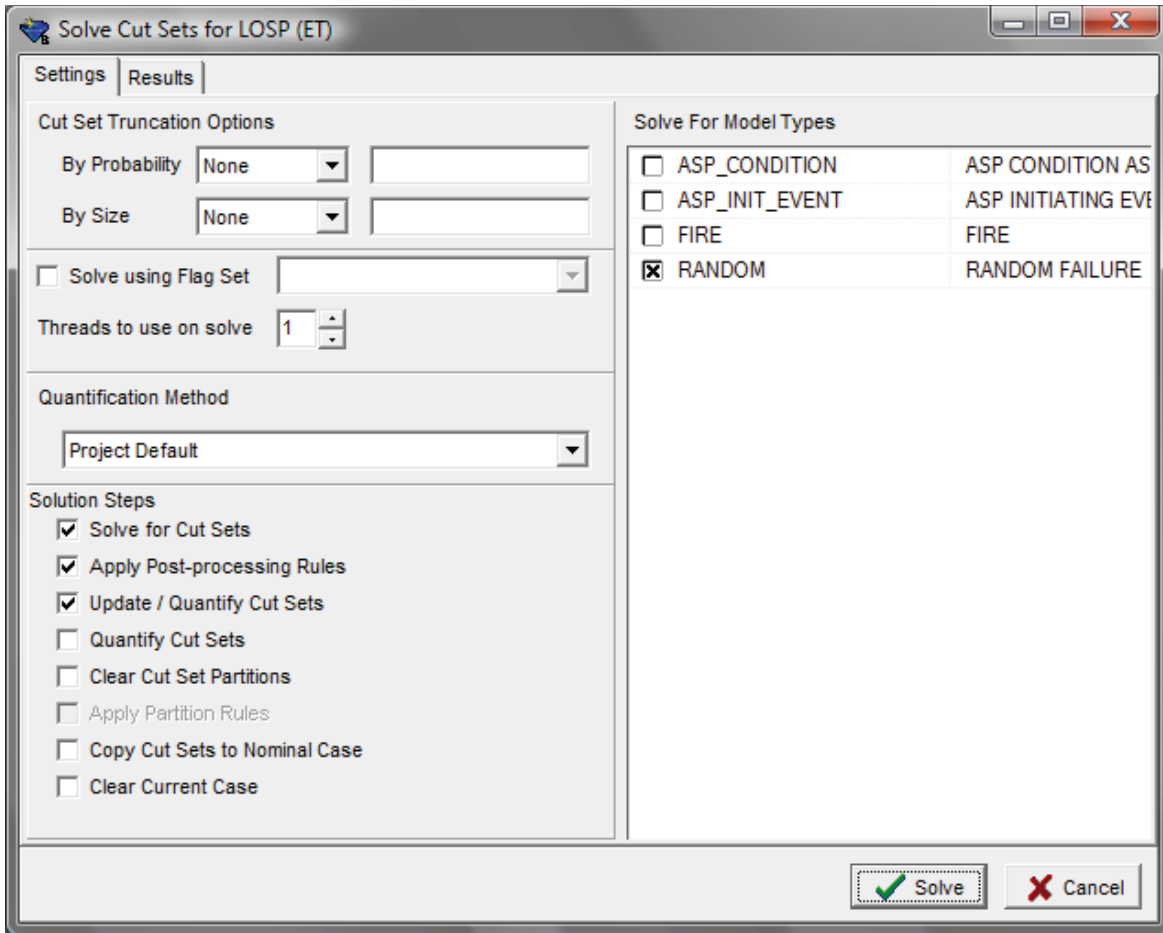
Figure 16. Model type selection for solving sequences

Report

There are multiple Report options, just like in SAPHIRE 7. The options from within the event tree menu (by right-clicking on an event tree) are:

- View Summary Results
- View Cut Sets
- View Importance Measures
- View Uncertainty

The other option is to select Publish from the project's main menu bar (see Figure 17 and 18). This report function allows the user to specify from a default listing of report outputs and select the document structure for printing.
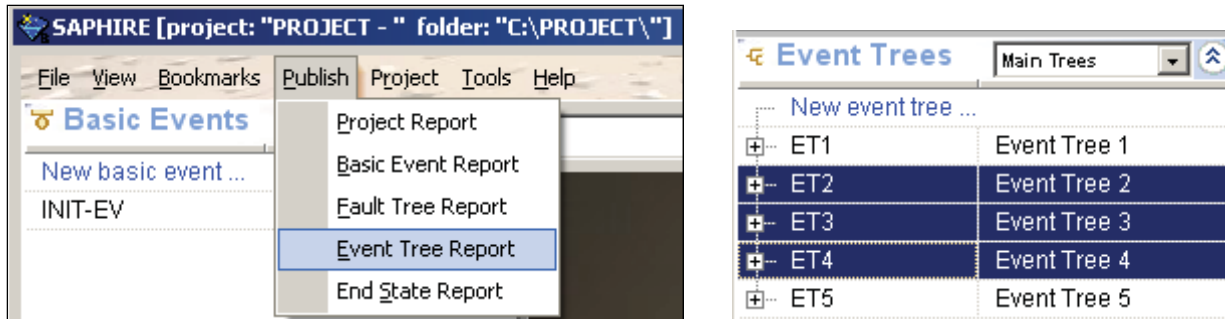
Figure 17. Event Tree Report path
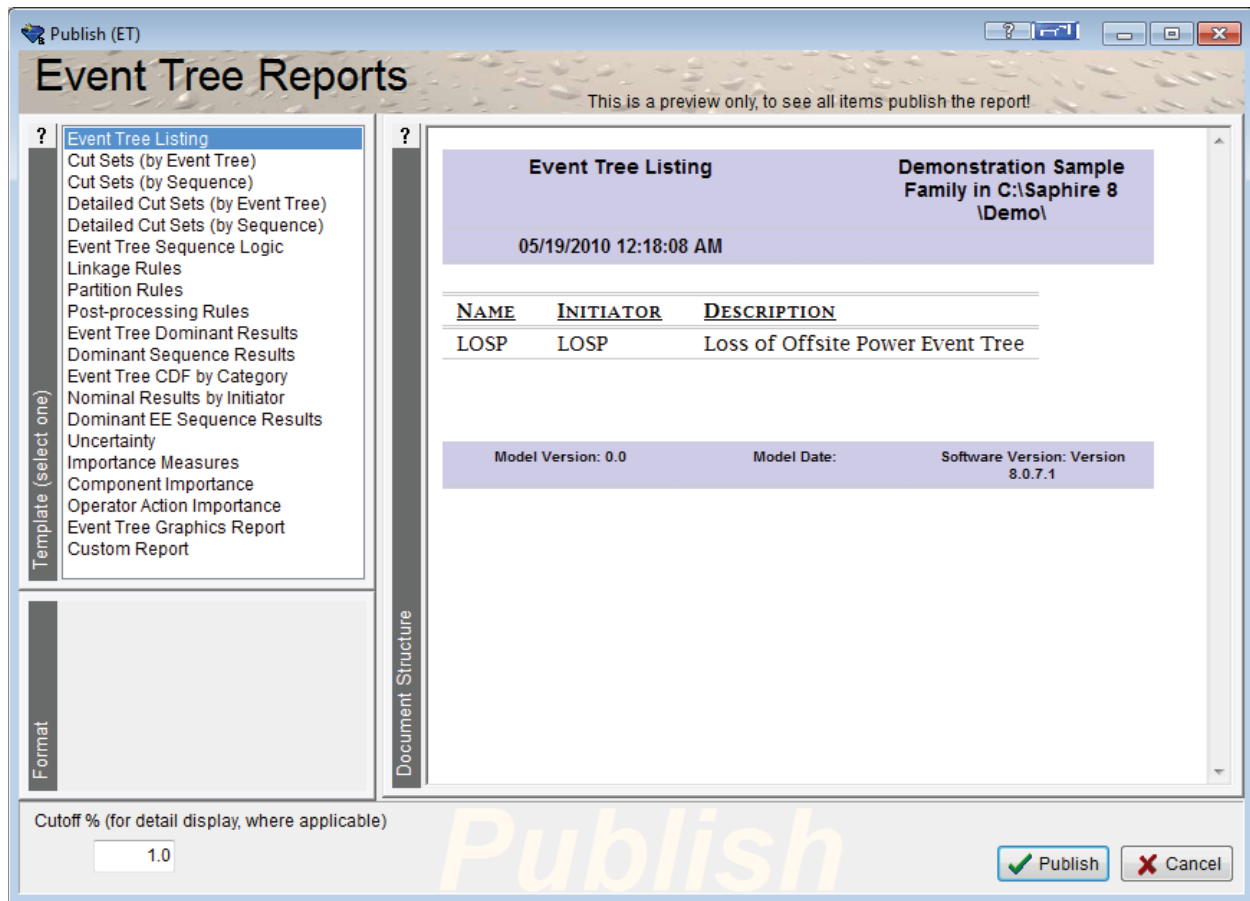
Figure 18. Event Tree report

2.3.2 Uncertainty Analysis

The uncertainty analysis is the same as that performed in SAPHIRE Version 7. The only difference is the menu options and improved plotting. In SAPHIRE Version 8, the Uncertainty analysis is performed in the **View Uncertainty** option after the cut sets have been generated. The type of uncertainty needs to be specified on whether all highlighted sequences are to be analyzed independently or if all highlighted sequences are to be analyzed as a group.

The different types of sampling available within SAPHIRE Version 8 are Latin Hypercube and Monte Carlo. To use either one of these sampling methods, its radio button must be selected along with the number of samples. The Uncertainty option may be selected for all highlighted sequences to be analyzed individually or to obtain a single result for the group of sequences.

Once the Uncertainty Method is selected then the Calculate button is pressed. SAPHIRE Version 8 will perform the sampling technique specified for the number of samples. Once SAPHIRE performs the uncertainty analysis it plots the results (see Figure 19, where we plot the uncertainty results for two sequences). The percentiles and overall mean can be displayed by selecting the **Results Table** button.
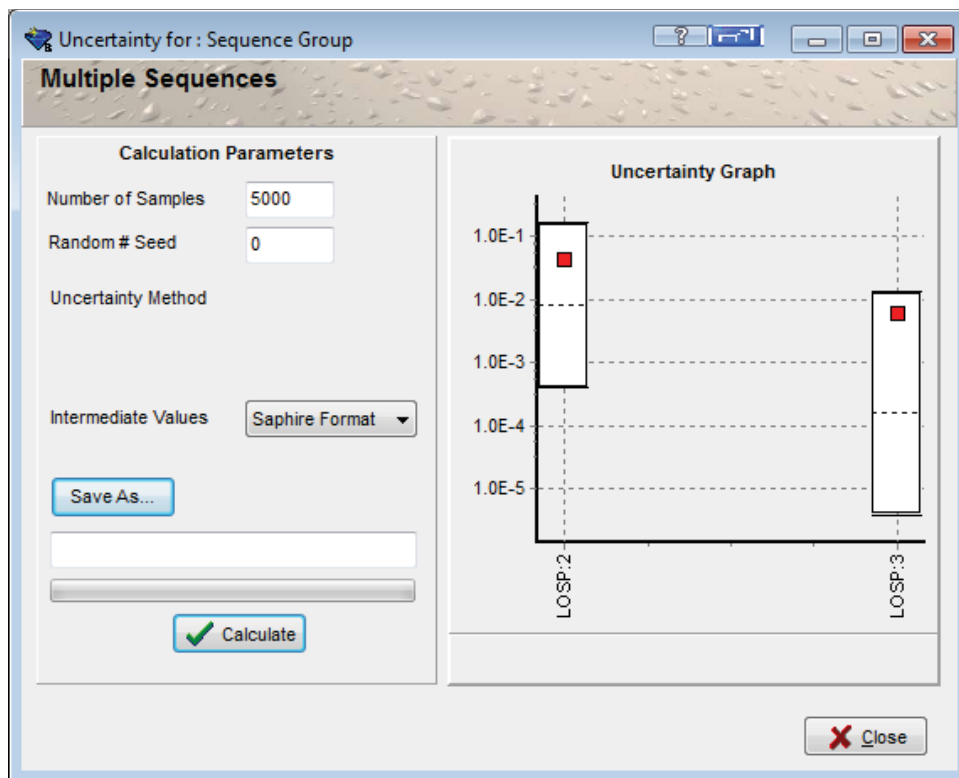

Figure 19. Uncertainty plot example

2.3.3 Importance Measures

The importance measures in SAPHIRE Version 8 are the same as in SAPHIRE Version 7. The only difference is the menu option and the type of outputs. The improvements on the Fussell-Vesely (FV) importance equation and on importance measure uncertainty analysis capability are applicable as well for event trees.

The importance measures can be reported as point estimates or the component variability can be incorporated into the results. The point estimate is calculated using the formal importance measure equations using just the input probabilities for the individual components and the overall point estimate for the accident sequence(s). However, if the importance of the components based on their variability is required, SAPHIRE Version 8 will perform the same calculation as stated for the point estimate except now it samples from the component's uncertainty and feeds this into the importance measure equation. For example, FV is calculated as

$FV_i = F_i(x)/F(x)$

where: F(x) is the original minimal cut set upper bound

  $F_i(x)$ is the minimal cut set upper bound with only the basic event of interest.

For the point estimate, the equation is solved only once for each component using their input probability. Now to take into consideration the component's variability, the FV equation is calculated for each sample. From the results of the samples, the FV mean, and other parameters are obtained for each component.

The report for the importance measure based on the point estimate is shown in Figure 20.
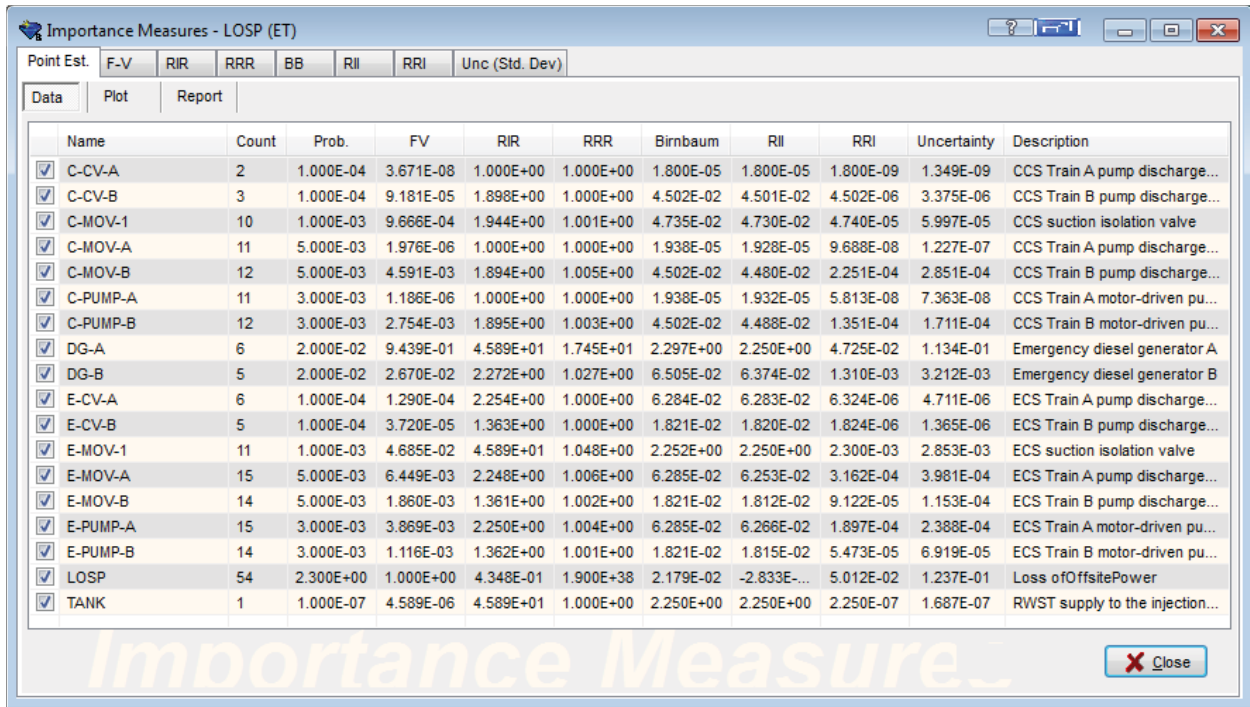
Figure 20. Importance Measures based on point estimate

The report for the importance measures allowing the component's variability to be propagated through is shown in Figure 21. The report shown below provides the results of the FV sampling.
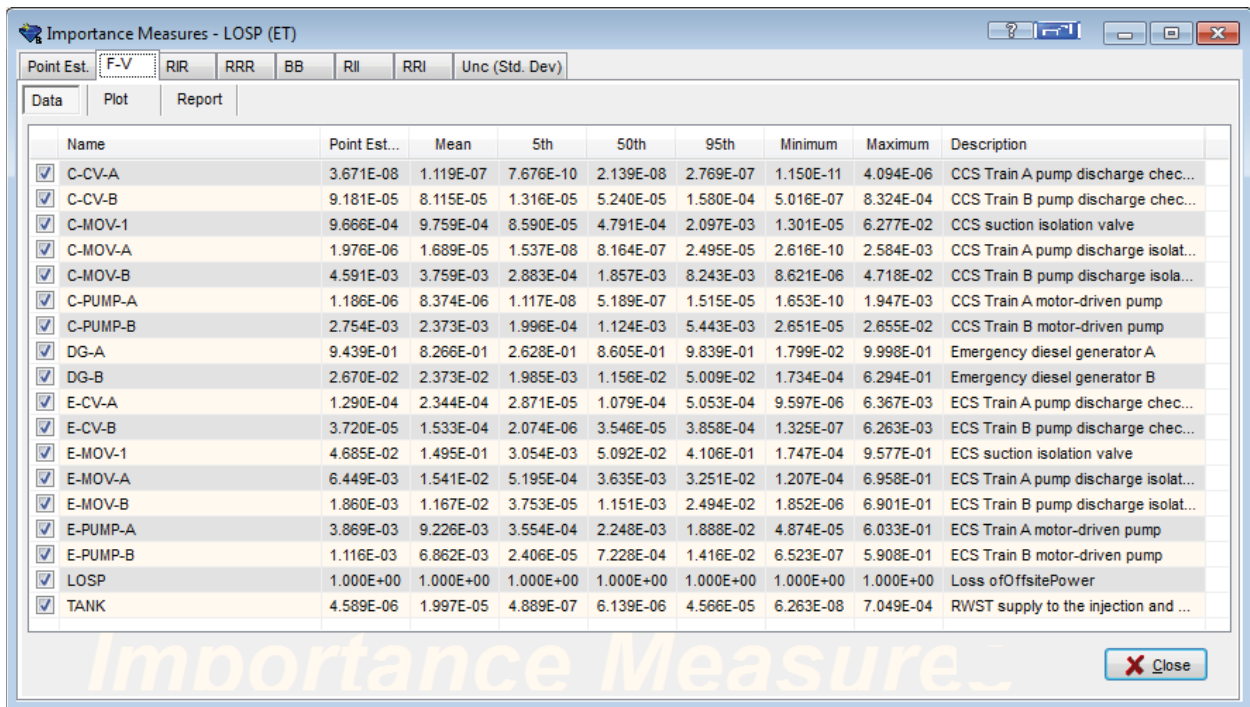

Figure 21. Fussell-Vesely Importance Measures report

The other report option for the importance measures is a plot of each component based on the sampling process and this plot is shown in Figure 22. In this plot, the 5th and 95th percentiles are shown via the white box, while the mean is shown as a small red box.
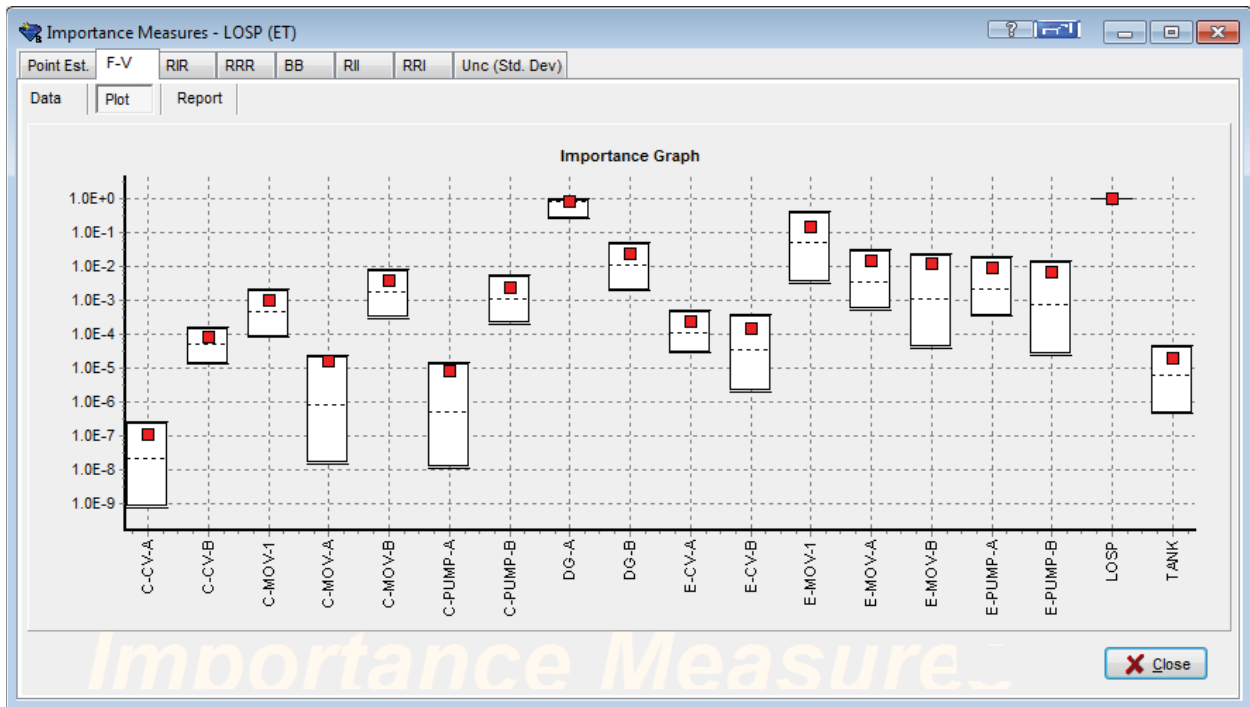
Figure 22. Plotting importance measure uncertainty

2.4   General Support Functions

General support function discussed in this section are:

- Search
- Bookmark
- Check Project
- Documentation

Search

SAPHIRE Version 8 has a search function that will return any object in the project that matches the search such as tree names, gates and basic events names, and including the matched text in the descriptions. It is not case sensitive. The results can be examined from the search for cross-references in the project, frequencies, quantification methods, and other information including a graphical view if viewing a tree with a movable magnified area.

As with any window opened, the search remains open and accessible in the main panel (see Figure 23). This capability is useful for large models.
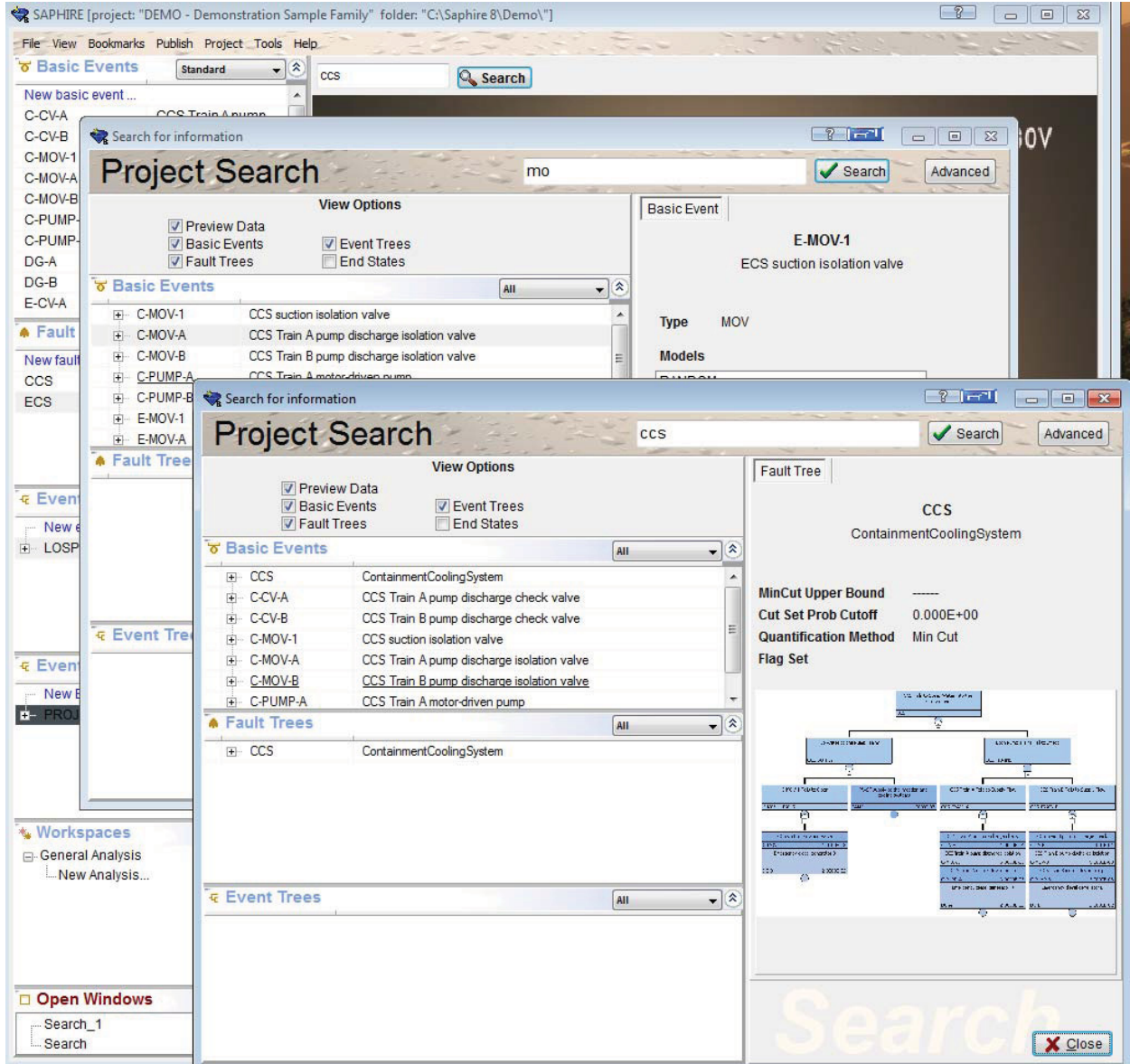

Figure 23. Sample search results

Bookmark

SAPHIRE Version 8 also has a bookmark option (see Figure 24) that provides a way to save a group of objects for later retrieval. To have all of the CCS basic events at the click of a

bookmark, the CCS are highlighted in the Basic Events window and a right click will bring up the option to bookmark the selection.
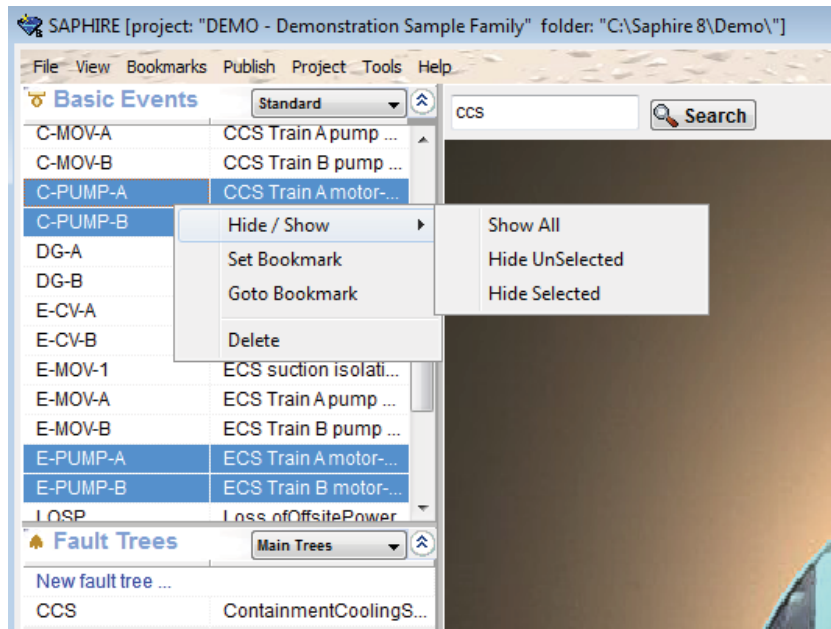

Figure 24. Bookmark example

Name and save the bookmark (see Figure 25) and it will be available for retrieval under Bookmarks. When the bookmark is retrieved, the previously highlighted basic events are selected.
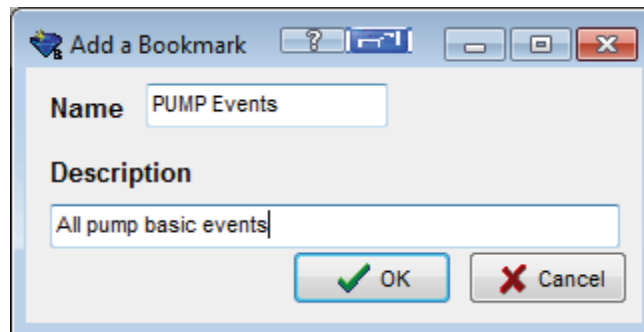

Figure 25. Bookmark add form

Check Project

A "Check Project" feature (found under the Tools option), shown in Figure 26, is also included in Version 8. This helps quality assurance checking of the PRA model database that was loaded into the SA user interface. Version 8 performs checks at the levels of the project, event, fault

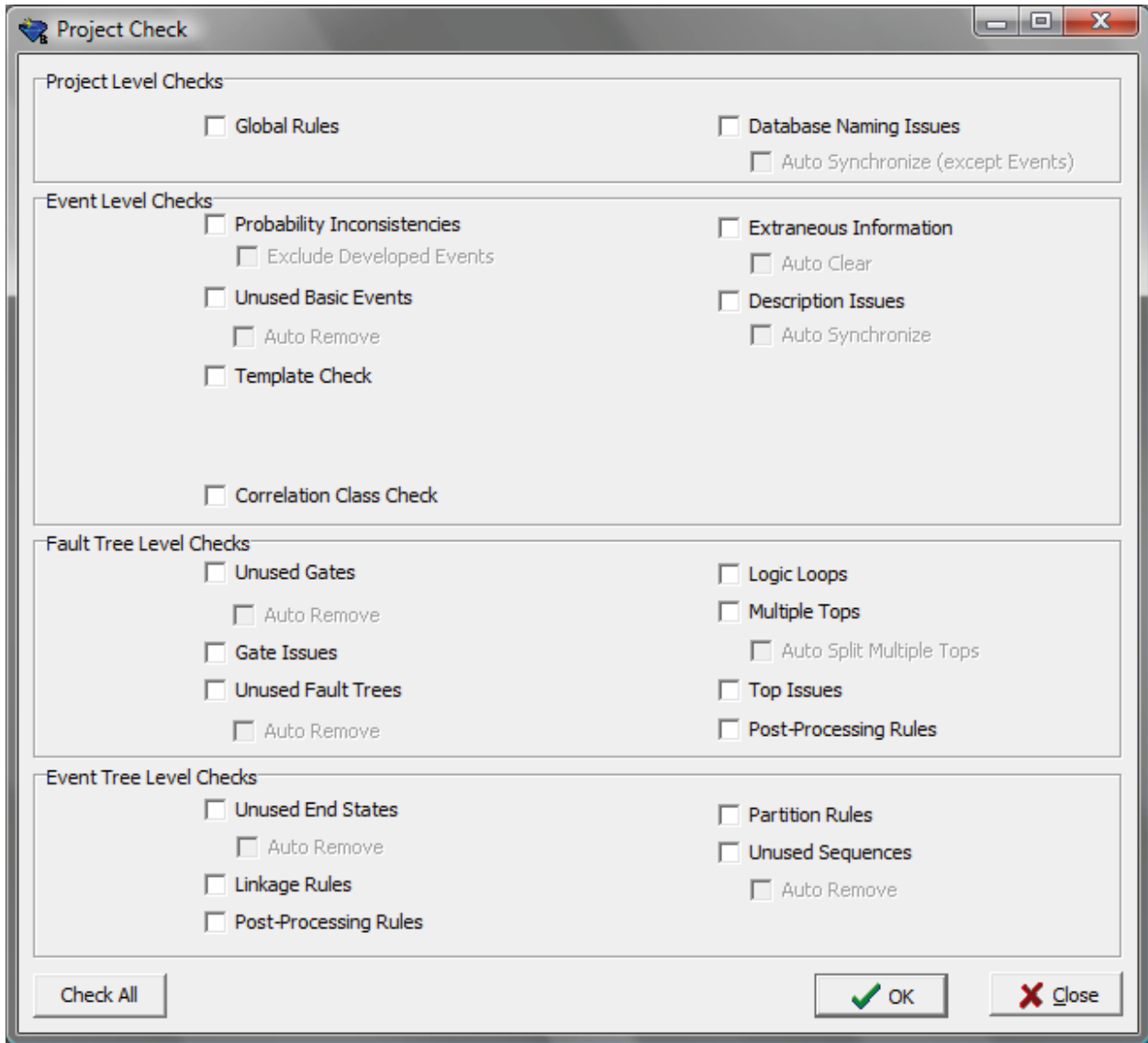tree, and event tree. The user should select which items to be checked, and then click the Ok button.

Figure 26. Check-Project feature

Documentation

On the toolbar, the Project → Documentation option (see Figure 27) provides additional information on basic events, event trees, and fault trees (name, description, notes, and references) as well as a way of adding tables to the SDP interface report.
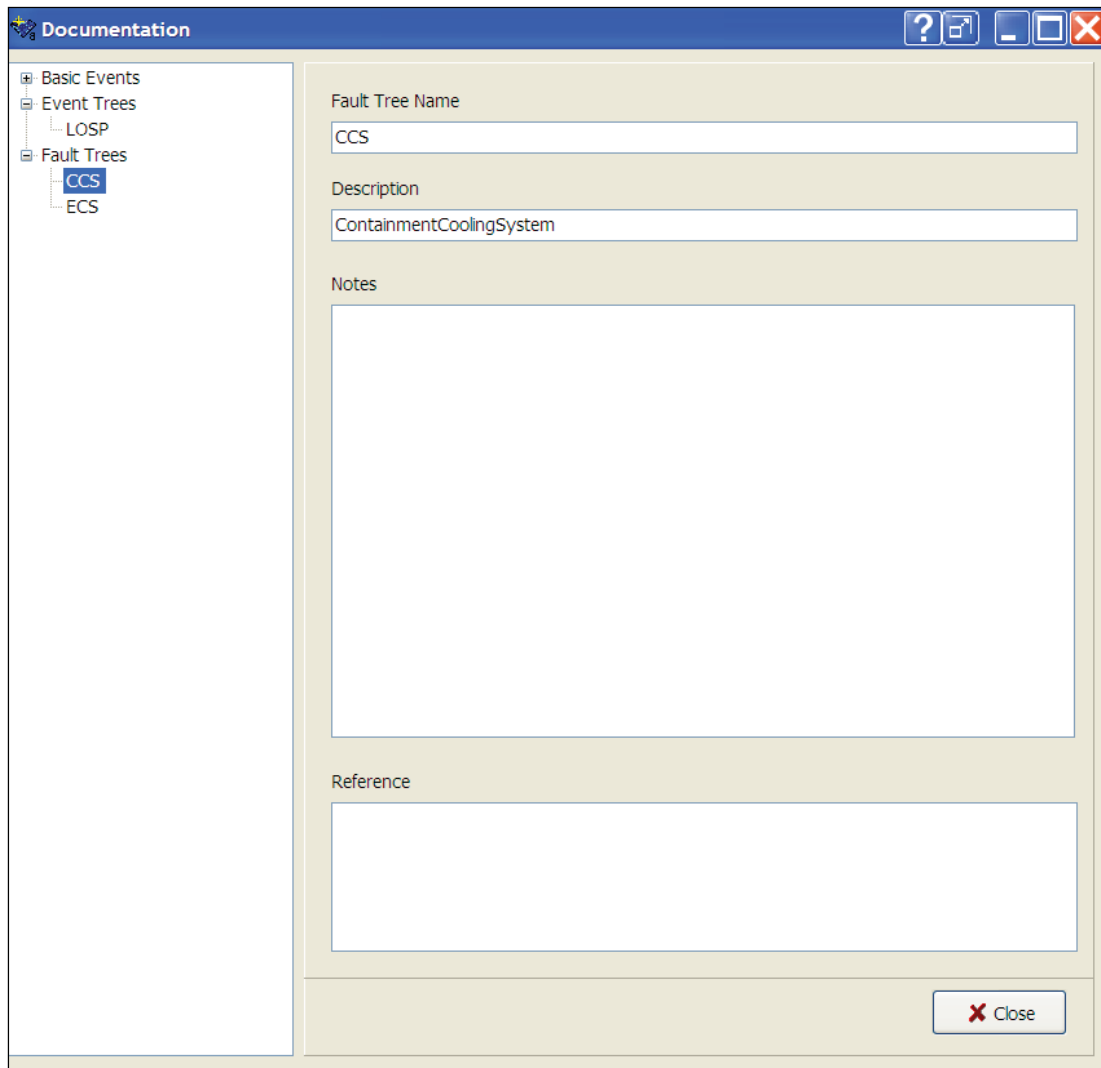
Figure 27. Documentation feature

Viewing Fault Tree and Event Tree Flag Sets

New in SAPHIRE is the display of flag sets assigned to fault trees and event trees. This information can be found in the SA interface by selecting a fault tree or an event tree, right clicking, and displaying View Summary Results.

3. MODELING FEATURES AND CAPABILITIES

SAPHIRE Version 8 has been designed to support improvements in SPAR model development and use. New features and capabilities were incorporated for external events and LERF/Level 2 PRA modeling. In addition, Version 8 was designed for working with integrated models.

3.1 External Events Modeling via an Accident Sequence Matrix

Development of models which make use of similar event tree structures, such as external events models, was made user-friendly by providing a method to import event tree logic and sequence flag sets. This method allows a model developer to specify the accident sequence information in a spreadsheet. An example input deck is shown in Figure 28.

---

* Model Type Name,Model Type ID,IE,IE Freq.,IE Desc.,Event Tree Name,Event Tree Desc.,X-fer to,End State Substitution,Flag Set Name,Flag Set Desc.,Flag Set Setting(s),,,,

FIRE,FIR,IE-FRI-1,4.84E-05,Fire Scenario 1,FIRE1,"Demo Fire scenario 1",LOSP,,FIRE_FS_1,Flag Set for Fire Scenario 1,E-CV-A

FIRE,FIR,IE-FRI-2,2.67E-04,Fire Scenario 2,FIRE2,"Demo Fire scenario 2",LOSP,,FIRE_FS_2,Flag Set for Fire Scenario 2,E-CV-B

FIRE,FIR,IE-FRI-3,2.58E-04,Fire Scenario 3,FIRE3,"Demo Fire scenario 3",LOSP,,FIRE_FS_3,Flag Set for Fire Scenario 3,E-MOV-1

FLOOD,FLI,IE-FLOOD-1,4.84E-05,Flood Scenario 1,FLOOD1,"Demo Flood scenario 1",LOSP,,FLOOD_FS_1,Flag Set for Flood Scenario 1,E-CV-A

FLOOD,FLI,IE-FLOOD-2,2.67E-04,Flood Scenario 2,FLOOD2,"Demo Flood scenario 2",LOSP,,FLOOD_FS_2,Flag Set for Flood Scenario 2,E-CV-B

FLOOD,FLI,IE-FLOOD-3,2.58E-04,Flood Scenario 3,FLOOD3,"Demo Flood scenario 3",LOSP,,FLOOD_FS_3,Flag Set for Flood Scenario 3,E-MOV-1

SEISMIC1,EQ1,IE-EQ-BIN-1,1.036E-03,Seismic Scenario 1,SEISMIC1,"Demo Seismic scenario 1",LOSP,,SEISMIC_FS_1,Flag Set for Seismic Scenario 1,E-CV-A

SEISMIC2,EQ2,IE-EQ-BIN-2,2.560E-05,Seismic Scenario 2,SEISMIC2,"Demo Seismic scenario 2",LOSP,,SEISMIC_FS_2,Flag Set for Seismic Scenario 2,E-CV-B

SEISMIC3,EQ3,IE-EQ-BIN-3,8.740E-06,Seismic Scenario 3,SEISMIC3,"Demo Seismic scenario 3",LOSP,,SEISMIC_FS_3,Flag Set for Seismic Scenario 3,E-MOV-1

Figure 28. Information for an accident sequence matrix file

In SAPHIRE, the accident sequence matrix file is loaded via **Project → Tools → Add Accident Matrix** from the main menu. Click the open button and locate the accident matrix file desired.

Click the Add button and the process begins. Information will appear in a text box as the accident matrix file is being processed. Any errors or warnings will also be displayed in this box. When done, click the OK button. The project will now be populated with the event trees and basic events described in the accident sequence matrix file.

This capability builds upon an existing model – it does not build a new project from scratch. In the above example an internal initiating events model is extended to include external initiating events.

SAPHIRE will create the following from the above line items:

Model Type: (Model Type Name, Model Type ID)

Initiating Event: (IE, IE Freq., IE Desc.)

Event Tree: (Event Tree Name, Event Tree Desc.)

End State Substitution (created in the linkage rules)

Flag Set: (Flag Set Name, Flag Set Desc., Flag Set Setting(s)), where the "Flag Set Settings" is the basic event name which will be set to True (the ASM does not currently set the basic event to False or Ignore).

The following must exist:

The transfer tree: (X-fer to)

3.2   LERF/Level 2 Models

SAPHIRE Version 8 features and capabilities were extended to support new LERF/Level 2 modeling requirements. SAPHIRE Version 7 supports LERF/Level 2 models which utilize Bridge Event Trees (BETs) and Containment Event Trees (CETs). While this BET/CET modeling capability remains in Version 8, requirements were specified to build LERF/Level 2 models which are compatible with "extended Level 1 SPAR models" and utilize containment phenomena event trees (CPETs) and BETs. Extended Level 1 SPAR model event trees have an end state corresponding to plant damage states (PDSs) rather than core damage (CD).

In Version 8, different end states, e.g., CD, PDS, LERF, can be modeled as different phases (Phase 1 = CDF, Phase 2 = PDS, and Phase 3 = LERF). The LERF/Level 2 model can be run in the SA interface just as it can be done in SAPHIRE Version 7. The model can also be run in the ECA and the GA interfaces. There are differences in performing the calculation in these interfaces versus the SA interface. First, the underlying calculation does not gather on the intermediate PDS end state (assuming the LERF end state of a LERF model is chosen to be analyzed). Gathering is only performed on the last chosen end state. Thus, results are

reported from the initiating event through the last chosen end state. Second, the GA interface allows a comparison between the new case and the nominal case only when basic event probabilities are modified. It should also be noted that LERF/Level 2 models are not run in the SDP interface (although the capability exists) since that interface has a specified method for estimating LERF.

Additionally, new keywords were developed to support LERF/Level 2 model building and are described in the Technical Reference Volume 2.

Phases

Phases can be assigned to the project by opening the Project menu and selecting Phases as shown in Figure 29. Phases are discussed further in Chapter 4.6.
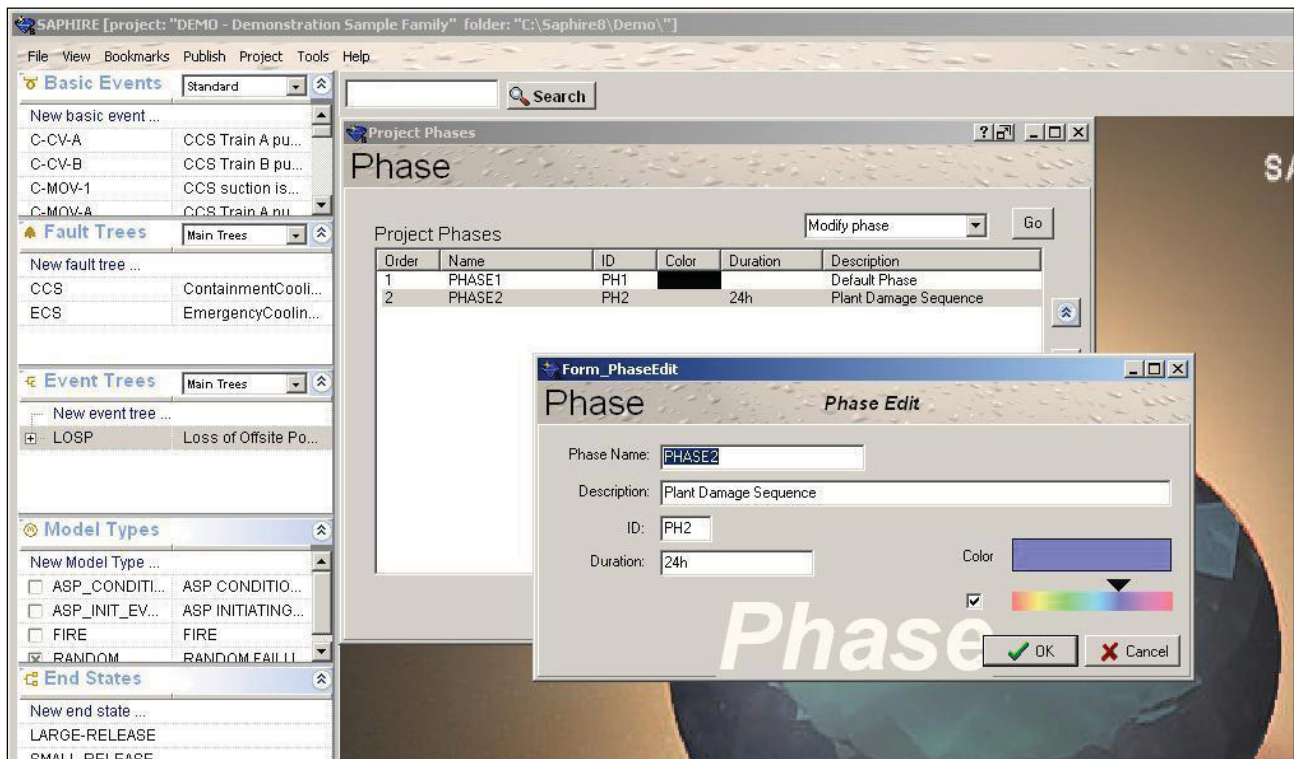

Figure 29. Adding a new Phase option

3.3   Integrated Models

Basic event "Applicability" is a way to separate a component into different failure types. For example, in addition to random failures, one may have failures related to earthquakes, high winds, or other conditions. The different failure environments may have their own failure models, parameters, descriptions, and uncertainty. The applicability option is used to control

phased-mission models, where one may have different failure models for different phases. In SAPHIRE Version 8, up to 32 different model types and 100 phases may be defined.

Unlike the "Analysis Type" used in SAPHIRE 7, any of the applicable model or phase-related events may appear in the cut sets in any combination, as dictated by the fault tree and event tree logic.

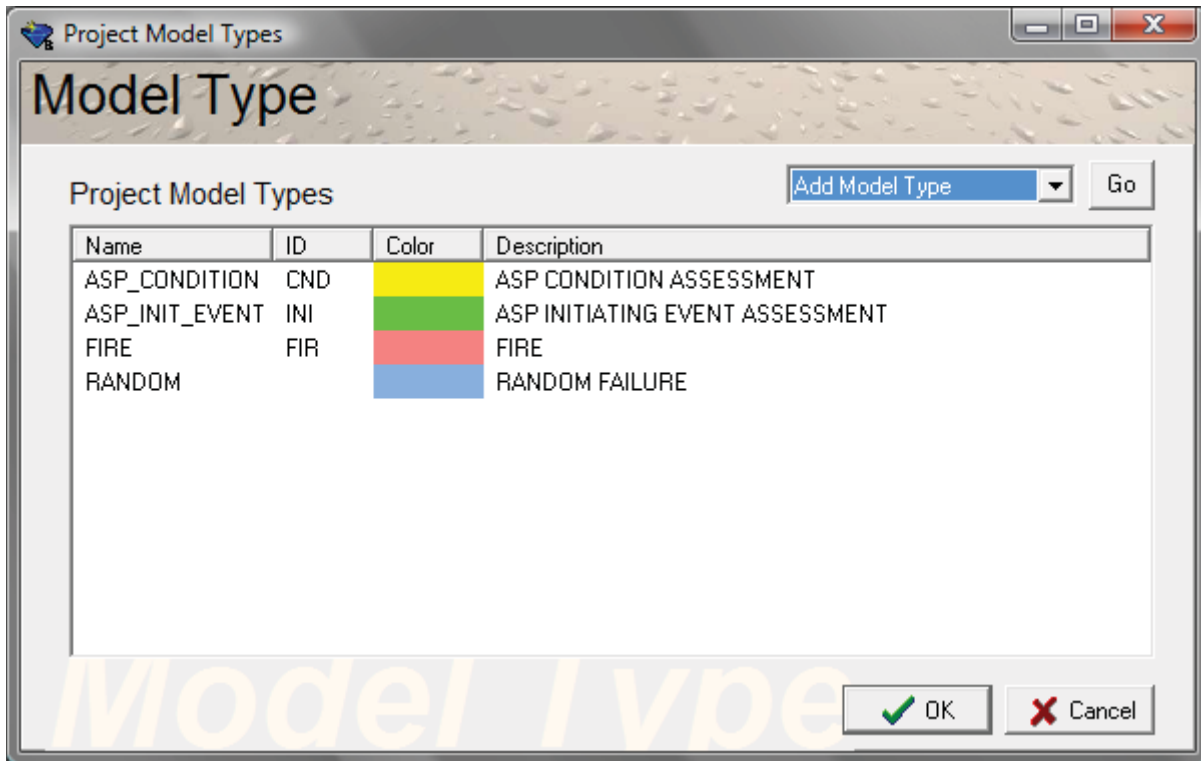Basic event model types are defined under the **Project → Model Type** option (see Figure 30).

Figure 30. Adding a new Model Type

Project phases are assigned as described previously, and will appear in a basic event's "Applicability" tab. A user will be able to evaluate a PRA for a specific model type (e.g., fire, flood, either fire or flood) in conjunction with different phases (e.g., Level 1 core damage, LERF).

Version 8 support for integrated models includes:

Building and modifying integrated models

- Assigning Model Type to Event Trees and Basic Events (Gates and Fault Trees are not assigned a model type)

- Assigning Phase to Event Tree sequences, Fault Trees, and Basic Events

- Comparing an imported model (or model section) to a model loaded into the Standard Analysis interface via the "Integrate Project" feature

- Creating models using the Accident Sequence Matrix capability

- Creating models in the SA interface

- Modifying models in the SA interface

- Modifying models in the GA interface

Viewing integrated models

- Event Tree sequence phases are color coded

- Fault Tree basic events are color coded and can be viewed using "View Expand"

Solving integrated models

- Solving any combination of Model Types in the SA, the SDP, the ECA, and the GA interfaces.

Reporting for integrated models

- Reporting results of any combination of integrated model solved

- Reporting results from the Initiating Event through the selected Phase end state for workspace calculations

4.   QUANTIFICATION ALGORITHMS AND METHODS

4.1   Single Pass and Multiple Pass Solving

SAPHIRE Version 8 has two options when solving the model:

- Single pass with cut set update option
- Multiple pass

SAPHIRE Version 7 solves only with the single pass with a cut set update option. There is one difference between the Version 7 and Version 8 single pass algorithm, namely, that the Version 8 algorithm will automatically adjust common cause failure compound basic events in a group when the group's component(s) are out for test and maintenance. For example, one component is out for test and maintenance, then the compound basic event common-cause failure probability will be automatically adjusted for a group of one less components.

The difference between the single pass with cut set update and the multipass solving routines are:

- The multiple pass solution algorithm will ensure that all sequence recovery rules are applied when basic events in the model are specified as a logical "True." The "single pass with cut set update option" does not do this (in Version 7 or in Version 8).

- The multiple pass algorithm will remove non-minimal cut sets if a basic event failure probability is modeled as 1.0 versus a logical "True." That is, the multipass algorithm automatically performs a cut set update. The cut set update performed in the "single pass" solution removes non-minimal cut sets, but it does not remove non-minimal cut sets that exist as a result of setting a basic event failure probability to 1.0 instead of a logical "True."

- SAPHIRE 8 performs the "base case" and the "new case" solving at the same truncation level for both algorithms. In SAPHIRE 7, which uses the single pass with a cut set update option, it is possible that these two cases could be solved at different sequence truncation levels resulting in calculating a delta between results truncated at different levels.

These two algorithms are outlined below.

SAPHIRE Version 8 follows the steps below for the "single-pass" solution:

1. Recalculate the "nominal case" to be able to determine the increase in risk.
   a. Generate basic event data with no change sets marked. If the no test/maintenance option is used, all Test and Maintenance (T&M) events will be set to zero.
   b. Solve all sequence cut sets using a predetermined (but low) truncation.
   c. Apply recovery rules to all sequence cut sets.
   d. If identified in the project constants, perform a cut set update to ensure non-minimal cut sets are removed.
   e. Re-quantify the cut sets.
   f. Store the cut set results via a base case update.

2. Calculate the risk increase.
   a. Generate basic event data with SDP changes indicated by the analyst. If the no test/maintenance option is used, all T&M events will be set to zero.
   b. Solve sequence cut sets using the same truncation for the "nominal" case.
   c. Apply recovery rules to all sequence cut sets.
   d. If identified in the project constants, perform a cut set update to ensure non-minimal cut sets are removed.
   e. Re-quantify the cut sets.
   f. Store the cut set results for use in determining the SDP annualized CDF.

SAPHIRE Version 8 follows the steps below for the "multi-pass" solution:

1. Recalculate the "nominal case" to be able to determine the increase in risk.
   a. Generate basic event data with no change sets marked. If the no test/maintenance option is used, all T&M events will be set to zero.
   b. Solve all sequence cut sets using a predetermined (but low) truncation.
   c. Apply recovery rules to all sequence cut sets.
   d. Perform a cut set update to ensure non-minimal cut sets are removed.
   e. Re-quantify the cut sets to ensure they have the proper values.
   f. Store the cut set results via a base case update.

2. Calculate the risk increase.
   a. Generate basic event data with SDP changes indicated by the analyst and store these values for use in Step 2g. If the no test/maintenance option is used, all T&M events will be set to zero.
   b. Change TRUE events to 1.0 events (to ensure application of recovery rules).
   c. Solve sequence cut sets using the same truncation for the "nominal" case.
   d. Apply recovery rules to all sequence cut sets.
   e. Change events having a probability of 1.0 to TRUE events (to ensure that non-minimal cut sets will be removed during the next step).
   f. Perform a cut set update to ensure non-minimal cut sets are removed.
   g. Re-quantify the cut sets so they have the proper values stored from step 2a.
   h. Store the cut set results for use in determining the SDP annualized CDF.

The SDP and the ECA "workspaces" discussed in Chapter 5 utilize the algorithms as given above. The GA workspace, however, does not first calculate a "nominal case." This is because changes to the model loaded into the GA workspace may include model logic changes in addition to probability changes of existing basic events which results in essentially a new model from the originally loaded model.

4.2 Risk Assessment Standardization Project (RASP) Common-Cause Failure (CCF) Module

The RASP CCF module uses Boolean reduction to realize combinations of CCF basic events, and removes approximations made in the conditional CCF probability modeling for events assessment as a result of setting components' failure modes to 1, TRUE, 0, or FALSE. The RASP common cause basic event is set as a new calculation type. The CCF type "R" can be found in the basic events under the "Failure Model" option as shown in Figure 31.
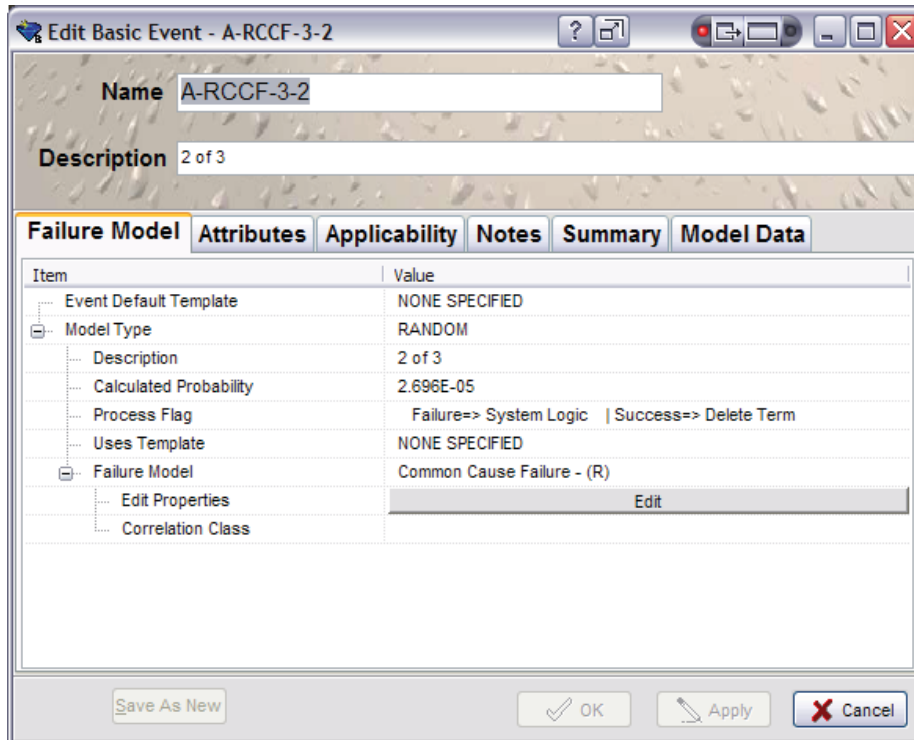

Figure 31. A basic event set up as the new CCF type

This CCF module is implemented in Version 7 and in Version 8; however, additional capability is included in Version 8. Version 7 will assign the RASP CCF probability to a single basic event that was specified to use the RASP CCF methodology. This probability is calculated by a fault tree containing all relevant component failure combinations. The fault tree is generated automatically by the RASP CCF module, and is solved to provide a single "rolled-up" CCF probability to the basic event of type R. Version 8 retains this capability, however, in addition, it expands the different combinations of common cause component failures which result in the "rolled-up" CCF probability (i.e., RASP CCF module cut sets which contribute to the "rolled-up" CCF will appear in the sequence cut sets) into the PRA model. When the contributing combination basic events are expanded, recovery rules can then be applied to specific combinations. Note, for high-redundancy level CCF events (e.g., five, six), SAPHIRE may generate a large number of CCF-related combinations.

When using the RASP CCF method, the user will select for type "R" basic events, a calculation type:

- "U" to indicate component is failed (unknown type)
- "D" to indicate a component is failed (dependent type)
- "Y" component is failed (independently)

The RASP module will determine the conditional CCF probability by ANDing the RASP CCF module fault tree with the basic event failure mode of interest (e.g., component A fails to start). Cut sets will be found from this tree, and will include terms involving different failure modes such as component A independent failure to start ANDed with components B and C failure to run due to CCF. These cut set results are then divided by the probability that A fails to start. The RASP CCF module will solve the conditional fault tree according to the type of failure specified. In addition to calculation types "U", "D", and "Y", the RASP CCF methodology also handles components in T&M, not failed, and basic events set to process flag Ignore. Process flag TRUE is treated as the new calculation type "U."

Also, in SAPHIRE Version 8, the RASP module includes a calculator to view the details of the CCF calculation (see Figure 32).
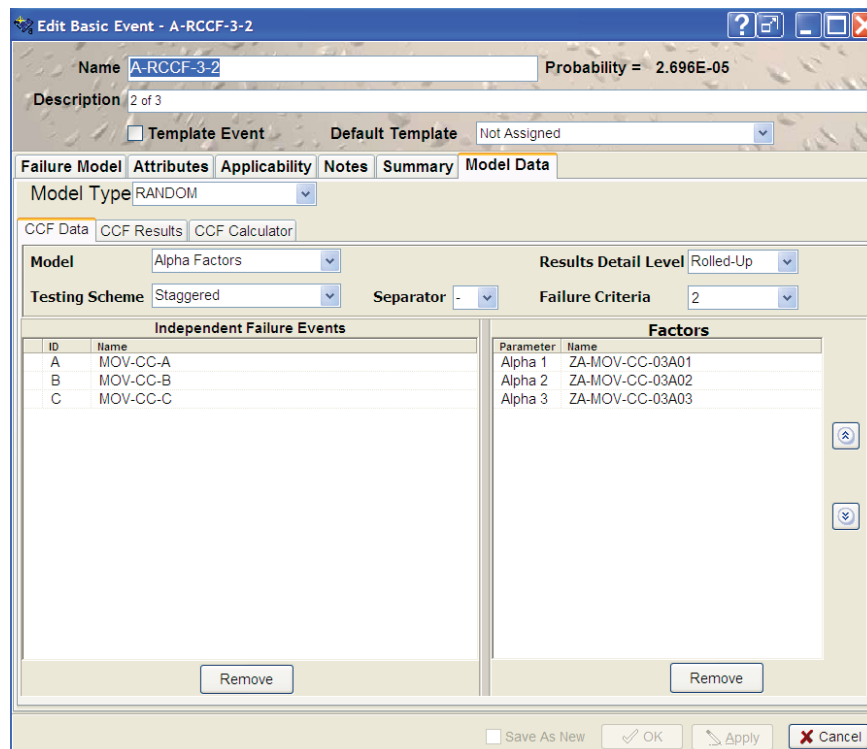

Figure 32. RASP CCF calculator

The settings for all common cause failure basic events (RASP CCF as well as compound) can be found in one place under the **Project → User Settings → RASP CCF** option (see Figure 33).
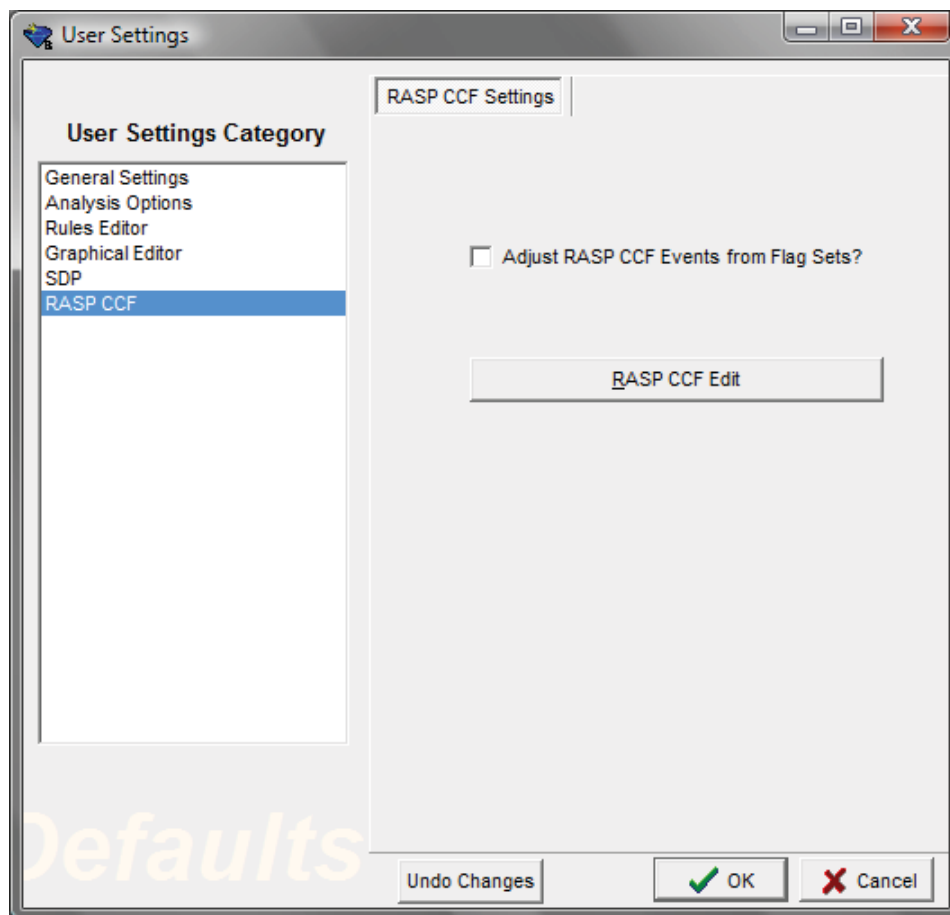

Figure 33. User settings for RASP CCF

Click on "RASP CCF Edit" to view the basic event information for CCF events (see Figure 34).
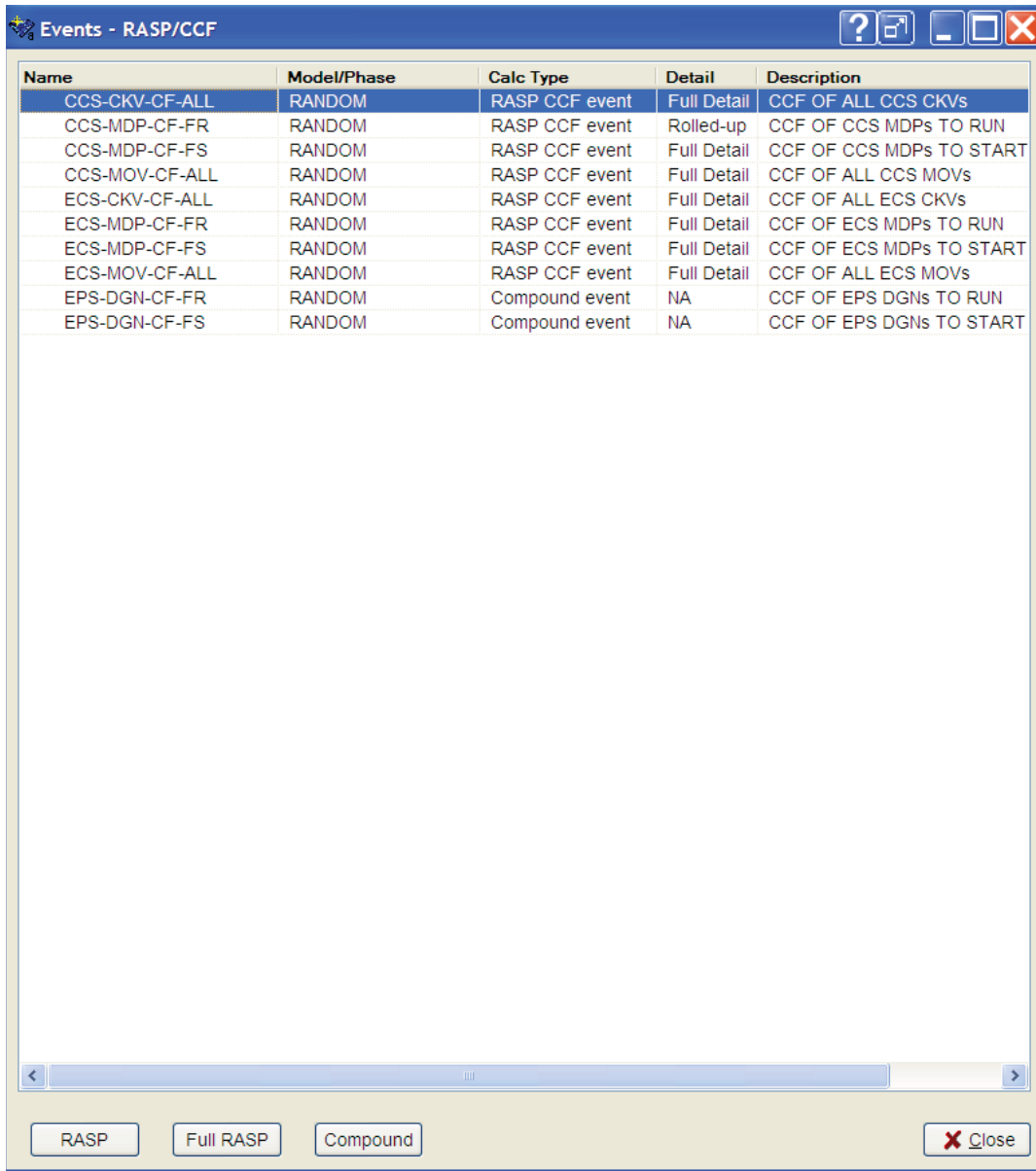
Figure 34. RASP CCF edit screen

Clicking on the "RASP" button will convert highlighted basic events to the new RASP CCF format (using the "rolled up" option). Clicking on the "Full RASP" button will convert highlighted basic events to the new RASP CCF format (using the "expanded" option). Clicking on the "Compound" button will convert highlighted basic events to the older-style CCF format (using compound events).

4.3   Flag Set Adjustment for RASP CCF Basic Event Probabilities

Flag sets are a special type of change set which are applied to specified sequences after solving fault trees and event tree sequences. Flag sets will modify a basic event to be True, False, or Ignore. When a flag set is applied to a sequence and a component is set to TRUE, system redundancy is reduced. However, the associated CCF probability is not automatically adjusted in Version 7 for the new common cause failure group (which would be one less, if one component is set to TRUE). In Version 8, CCF probabilities are automatically adjusted in the affected sequence when a flag set reduces a redundancy level associated with that CCF event. This enhanced capability removes the potential to underestimate CCF probabilities when using sequence flag sets. Note thought that this enhanced CCF probability adjustment can be use only for the RASP CCF method for basic events specified with calculation type R (not the older compound CCF basic events which are of calculation type C). To utilize this capability, mark the option's checkbox under the **Project → User Settings → RASP CCF** option.

4.4   New Rule Capabilities

Rule/basic event checks

Version 8 performs automatic checks of the user defined rules and the PRA model basic events. If basic events in the rules are not found in the model, SAPHIRE will alert the user.

Nesting

An example of the linking rule structure is:

```
If  "some search criteria"  …  then

Endif

Save_Rule_Results

| Next section of rules…would apply to the results of the previous rule
section

If  "some search criteria"  …  then

Endif
```

When SAPHIRE reaches section 2, the rules from the first section will be "applied" in that those rules will be complete and the rules in the second section will be able to search on the newly-applied rules from section 1.

Rule Layering

This feature extends the default linking rule behavior and allow linking rules to be applied from a "last" to "first" approach, where "last" would be those rules in the event tree closest to the end of the sequence.

For example, if the transient event tree has the rule:

```
| TRANS link rule

if always then

  AFW = AFW1;

  RPS = RPS-FT;

endif
```

and a fire event tree (that transfers to the transient event tree) has the rule:

```
| FIRE link rule

if always then

  AFW = AFW-FIRE;

endif
```

The rules closest to the end of the sequence (those in TRANS) would be applied first, yielding:

AFW = AFW1

RPS = RPS-FT

Then, the next closest to the end would be applied (those in FIRE), leaving:

AFW = AFW-FIRE

RPS = RPS-FT

In SAPHIRE Version 7, the rules AFW = AFW-FIRE and RPS = RPS-FT would not have been applied. But they will be in Version 8. In addition, the rule layering will be applied to all applicable trees in the model – the user can not pick and choose to apply or not apply the rule layering capability to only certain applicable trees.

Keywords

Some key words were added to support the development of the LERF model design. Keywords for LERF model design are given in the Table 2.

Table 2. Keywords for LERF model design.

Keyword	Description	Comments

System(Fault-Tree-Name) = Flag(Flag-Set-Name);	Assigns the flag set named "Flag-Set-Name" to the fault tree "Fault-Tree-Name."	Similar to the existing sequence flag set assignment approach, for example:
		
		if ECS then
		
		  eventree(LOSP) = flag(FLAG-SET-1);
		
		endif

Sequence = Endstate(ES-NAME);	Assigns the current sequence the end state named "ES-NAME."	The existing rule approach is: eventree(LOSP) = endstate(ECS-END); Both the new and old approach will be valid.
Node(Fault-Tree-Name) = Phase(PHASE_A);	Assigns the node under the top event named "Fault-Tree-Name" for the current sequence to the phase named "PHASE_A."	The phase must already exist in the project before it can be used in a rule.
SeqTransfer(EXISTING_TRANSFER) = EventTree(NEW_TRANSFER);	Replace the transfer named "EXISTING_TRANSFER" to "NEW_TRANSFER."	The existing rule approach is: eventree(ORIG-TRAN) = eventree(NEW-TRAN);
If Endstate(ES_Name) … then	Ability to search on an end state.	Not Dynamic – if ES changed in rule then original ES is used until the next Rule Section
If System(SYSTEM_Name) … then	Ability to search on a top event.	Same as If System_Name … then
If Phase(PHASE_NAME) … then	Ability to search on a phase.	Not Dynamic – if Phase changed in rule then original Phase is used until the next Rule Section
SkipSequence;	Ability to stop linking the current sequence and not add it to the database.	The existing rule approach is:

Keyword	Description	Comments

		if "search criteria" then
		
		  ET-FT = Skip(ET-FT);
		
		endif
		
		Both the new and old approach will be valid.

SkipSystem(SYSTEM_Name);	Ability to skip (i.e., ignore) the top event named "SYSTEM_Name" in the current sequence.	
VarStr(VARIABLE_NAME) = "Low";	Ability to assign a text to a sequence-level variable.	Assignment can only be a string value or a different VarStr. No operators can be used to combine.
VarNum(VARIABLE_NAME) = 40 * 2;	Ability to assign a value to a sequence-level variable.	Assignment of a variable number can use the following numeric operators between numeric values or other VarNum()s. +, -, *, /, ^
If VarStr(VARIABLE_NAME) = "Low" … then	Ability to test a string variable in an if statement	Comparison "=" only and must be encased in () if adding other comparisons (ie. "*", "+", "/")
If VarNum(VARIABLE_NAME) < 40 … then	Ability to test a number variable in an if statement	Comparison operators only and must be encased in () if anding(*) with other comparisons <, >, =, <=, >=
End_Rule_Section;	Ability to save the results of the block of rules above this keyword (up to the previous Save_Rule_Results keyword, if present).	

Advanced Recovery Rules

Note that the advanced recovery rules are deprecated in SAPHIRE Version 8 due to the discontinuation of the Modula-2 compiler (this compiler was used to compile the advanced rule structure).

Automatic Updates of Linking Rules

SAPHIRE 7 did not update information in the linking rules when a basic event name was modified. SAPHIRE 8 does this automatically. A number of basic event names have been modified in the recent past in an effort to make basic event names more consistent and to prepare for the transition to SAPHIRE 8. The recompiling of these rules in SAPIRE 8 will identify errors in the rules (due to name changes or basic event deletion) that must be corrected.

4.5   Test and Maintenance

Test and Maintenance Set to Zero

For some applications, T&M basic events are removed from the model by setting their probabilities, typically an average probability, to zero. This gives a model baseline without test and maintenance. Within a certain interval of time, some systems, structures, or components may be taken out of service. By setting their T&M basic events to True and the other T&M basic events set to zero, the risk increase can be evaluated for real time T&M configurations.

In SAPHIRE 7, this capability is available by creating a class change set. In Version 8, the zero T&M baseline evaluation can be done by selecting a check box in the user interfaces. The zero T&M check box capability was included in the ECA and the GA user interfaces. In the Standard Analysis interface, the class change set can be used for this. The SDP user interface does not have a default zero T&M option since the SDP evaluates risk increases from an average T&M baseline model; however this option can be turned on under **Project → User Settings → SDP**.

Test and Maintenance Set to True

When a test and maintenance basic event is set to True via a change set, Version 8 will automatically adjust CCF compound basic events which use this basic event. For example, if a component belongs to a group size of six, the CCF probability for a group size of five will be used when the component T&M is set to True. In SAPHIRE 7, reducing the component group size would have been a manual user responsibility by choosing the appropriate CCF plug in for the affected compound event(s).

4.6   Phase Analysis

Version 8 includes limited phase analysis capability. One or more phases can be assigned to basic events (fault tree gates are not assigned a phase.), and event tree sequence nodes can be assigned one phase. Different nodes under an event tree top event can be given different phases.

Phase analysis features and capabilities completed are:

- Store phase name

- Store phase ID

- Store phase description

- Store phase duration

- Sore preceding phase

- Store phase color

- View, modify, add, or delete the general phase information (e.g., name, ID, color)

- A phase that is unused may be deleted

- A total of 100 phases will be available within a single project

- Allow individual basic events to be assigned (via "applicability features) to specific phases

- Allow event trees to be tied to one or more phases by modifying the nodal information. The phase assignment must proceed chronologically through the accident sequence.

- If a node does not have a phase assigned to it, the last phase found in a preceding node is assumed to be applicable. The default phase assigned to each node will be a system generated "PHASE 1" category.

- Generate phase-aware cut sets for a combination of success and/or failures resulting in a valid scenario.

4.7   Integrated Model Solving

Some PRAs require modeling failures due to causes other than random, such as fires, floods, or seismic events. Historically, model types other than "Random" were created using BETs (e.g., external events models) in SAPHIRE, and were run as separate models. Version 8 uses the Model Type definitions to create models other than "Random." In Version 8, the basic events have been extended to allow the user to specify a basic event's model type(s) as well as different reliability models for each applicable model type, and will display this information in one "virtual" basic event (A "virtual" basic event is one basic event that has multiple failure mode attributes such as fails due to a fire cause, fails due to a flooding cause, fails due to a random cause, etc.). Version 8 will allow combinations of model types to be run at the same time, and presents cut sets which can contain basic events of different model types (e.g., Fire + Random). In Figure 35, SAPHIRE automatically appends the user-defined suffix "-FIR" for fire model type basic events that was specified in the accident sequence matrix file.

Figure 35. Cut sets containing Fire + Random model type basic events

Additionally, if the accident sequence matrix feature is used for building an integrated model, then event tree sequence flag sets can be specified in the accident sequence matrix spreadsheet and these flag sets would be applied at the time of solving.

The solving of the fault trees and event trees is transparent to the user, which improves user checks of the PRA logic and model solving. For example, the expanded view of the fault tree (double click the fault tree, and select File → View Expanded) allows tracing the logical relation of system basic events which may be subject to random failure or other conditions. Also, flag sets for fault trees and event trees can easily be viewed.

Integrated models in SAPHIRE Version 7 may also be built using a different approach other than BETs. In this approach, event trees for different initiating events are created, and flag sets are used to specify which components are failed due to the initiating event, or conditions resulting from the initiating event. All basic events in this approach are specified as type "Random." Version 8 is able to solve models using this approach with the click of a button (see Figure 36). The buttons run macros for portions of the model, such as "Internal CDF Only," "Internal and External CDF," etc.
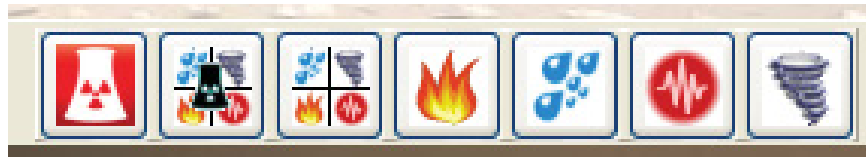

Figure 36. Macro buttons for integrated models

In such models, the basic events following different initiators retain their "Random" type reliability model. This is because the same fault tree used for "Random" initiating events is used in other types of event trees (e.g., Fire).

If the Version 8 "Model Type" feature is used to build integrated models, then basic events following different initiators can have different reliability models assigned. Consider, for example, a fire event tree which transfers to a loss of offsite power event tree. System failures in the loss of offsite power event tree could be due to fire-related only, random, or fire-related + random events. The random failure of a component could be given one type of reliability model, and the fire-related failure of the same component could be assigned a different reliability model. It may also be necessary to model fire sequence-specific component failure probabilities. This can be accomplished by assigning different fault trees to different sequences. This approach would result in assigning sequence specific fire-related component failure probabilities prior to solving. Rules could also be used to change the basic event's probability, but this would be a post-processing type of calculation.

To run models using the Version 8 model type feature in the SA interface, simply highlight any combination of event trees, right click and select "Solve." To run the model in the SDP, ECA, or

GA interfaces, select the Model Type check boxes to be solved as the analysis is stepped through.

4.8   Significance Determination Process Calculations

Screening Large Early Release Frequency Calculation

The SDP user interface requires the evaluation of the frequency of large, early release (LERF). The SDP method of estimating an increase in LERF is to apply a conditional LERF probability to relevant core damage sequences. This method was implemented into the SDP user interface. Under "User Settings" a list of the Level 1 PRA sequences can be drawn up. For each relevant sequence a conditional probability for LERF can be entered. After solving for minimal cut sets, the LERF multiplier factors will be applied. The SDP user interface reports the results as an increase in screening LERF to distinguish it from a frequency calculated with a LERF PRA model. The SDP user interface is the only one with this capability.

Duration Sensitivity Analysis

The SDP interface provides a sensitivity analysis graph (Figure 37) on the duration of the condition. It shows when risk-significant thresholds would be reached (assuming a "month" of 30 days).

Using the Demo-SDP project and failing the CCS Train A MDP, the following results are generated:
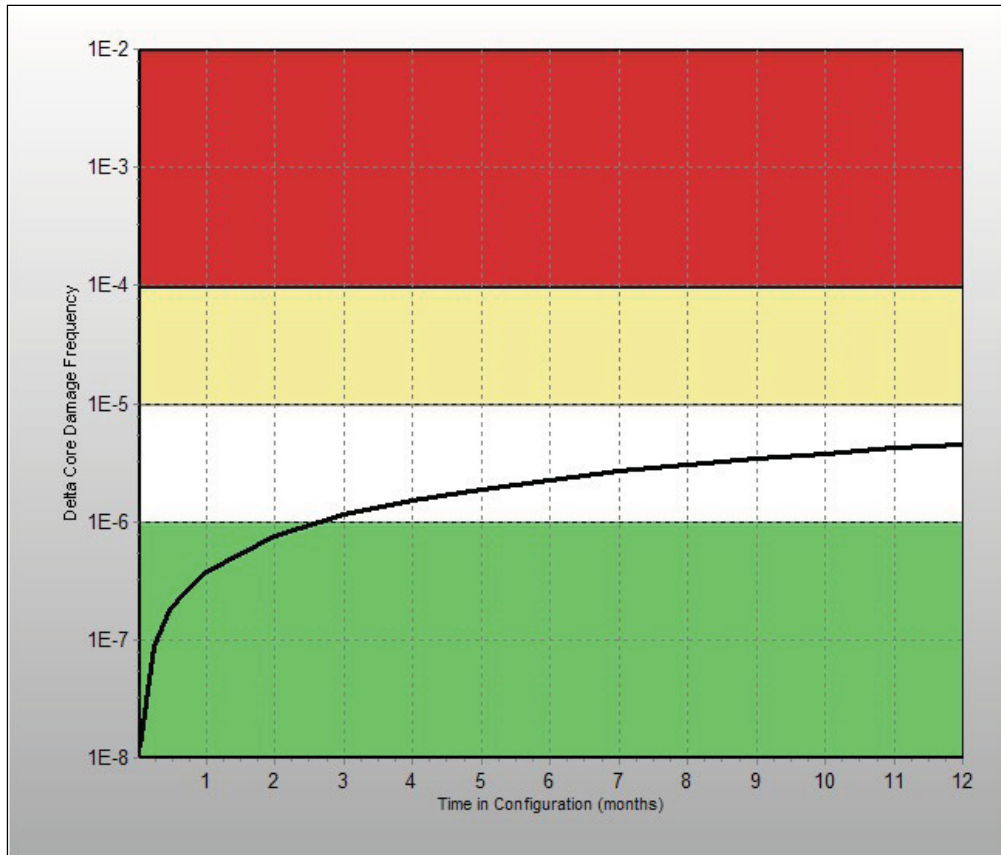
Figure 37. Duration sensitivity analysis curve

Underlying Model Calculation Adjustments

Underlying calculation manipulations for the SDP interface are:

- When a basic event is set to TRUE, SAPHIRE will model the condition as having a potential common cause failure

- When a basic event test and maintenance is selected, SAPHIRE will: 1) not include the failure to start and failure to run modes in the results (these failure modes are not applicable), and 2) will adjust the common cause failure probability to correspond to a common cause failure group of one less.

4.9   Events and Condition Assessment Uncertainty Analysis

Two general calculation types are performed, initiating event assessment and condition assessment. For the former, a single metric, the conditional core damage probability (CCDP), is calculated. This metric is used in an absolute sense in that its numerical value is not subtracted

64

from a nominal value to obtain an increase in risk. However, the condition assessment calculation requires two metrics, a CCDP and a core damage probability (CDP). In this calculation, the risk increase (over the time of the condition) is calculated.

Historically, only a point estimate value was calculated for the condition assessment CCDP and CDP. These two values were then subtracted to find the risk increase. However, a variety of complications arise when this approach is used, specifically on the change in risk measure, or "event importance" (Ie) which is given by the relationship

Importance$_{event}$ = CCDP – CDP = Ie

where: CCDP = the conditional core damage frequency (CCDF) duration of the condition

CDP = the core damage frequency (CDF) duration of the condition

The uncertainty results are based on sampling from the component's variability and then solving for the CCDF and CDF based on the sampled probability. SAPHIRE then subtracts CDF from CCDF to obtain a sampled ΔCCDF, which is multiplied by the duration. SAPHIRE stores this calculated value and continues this process for the number of samples noted. Once SAPHIRE has performed the number of specified samples, it orders the ΔCCDP values and pulls out the 5th, 50th, and 95th, then calculates the mean by summing up the ΔCCDPs and divides it by the number of samples.

The algorithm used to determine the uncertainty on the Ie using either Monte Carlo or Latin Hypercube sampling is given below:

1.   Solve all cut sets (to the project truncation level) for all sequences. Two lists are solved and stored, one for the nominal CDP case and one for the condition CCDP case.

2.   Create a list of all basic events appearing in cut sets for either the nominal or condition cut sets. There will be a total of J number of basic events.

3.   Start the uncertainty sampling loop.

        For I = 1 to N, where N is the total number of iterations specified by the user.

        a.   Go through the list of J events. For each event, obtain a random sample. If events are correlated, obtain a single value for all events in that correlation group. If an event is modified for the condition case, it will appear in the list twice, once to be used for the nominal (CDP) case and once for the condition (CCDP) case, with its values set for the respective case.

        b.   Calculate CDP$_I$, where CDP = 1 – exp(- λ_{CDF} * Duration). The λ_{CDF} is the value of the core damage frequency from the nominal case cut sets quantified using the samples obtained in Step 3a.

c.   Calculate CCDP$_I$, where CCDP = 1 – exp(- λ$_{CCDF}$ * Duration). The λ$_{CCDF}$ is the value of the core damage frequency from the condition case cut sets quantified using the samples obtained in Step 3a.

d.   Calculate Ie, where Ie = CCDP$_I$ - CDP$_I$.

e.   Store Ie, CCDP$_I$, and CDP$_I$ into their own respective array, where each array is N elements long.

Repeat the **For I** loop N times.

4.   Once the **For I** loop is complete and the three results arrays (Ie, CCDP, and CDP) are populated, sort each array.

5.   From each result array, determine the moments and percentiles.

This uncertainty analysis is performed in both Version 7 and 8. The graph of the uncertainty analysis, as shown in Figure 38, is added to Version 8 reports in the ECA interface. It color codes the percentages of the probability distribution function between the orders of magnitudes.
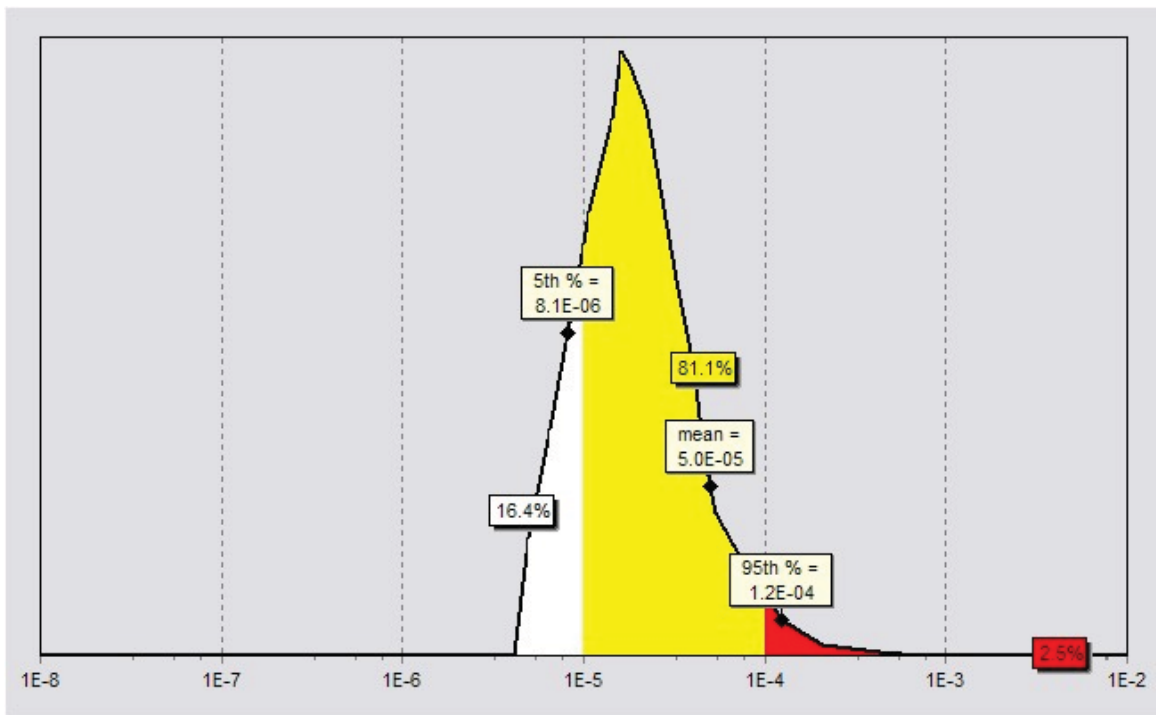

Figure 38. Graph of ECA uncertainty analysis

4.10   Gathering End States

LERF/Level 2 models can be run in the ECA and the GA user interfaces as noted previously. These models may also be run in the SA user interface in the same manner they are run in the SAPHIRE screen of Version 7. The SDP interface uses a different methodology to estimate LERF, although it, too, has the capability to run these types of models.

In the workspaces, the LERF end state can be quantified without stepping through numerous steps as done in Version 7. Workspace quantification does not gather on the intermediate Plant Damage States (PDSs) rather, cut sets from the initiator through the LERF end state are merged (and minimized). Gathering is performed on the last phase (equivalently the end state) defined. In the workspaces, partition rules other than gathering on intermediate end states (e.g., PSDs) can be applied. Gathering on PDSs in the workspaces is not performed unless the user specifies the PDS phase as the last phase in the model.

4.11   Calculation Capabilities from SAPHIRE 7

Calculation capabilities carried over from Version 7 into Version 8 include:

- Basic event calculation types
- Change sets
- Flag sets
- Single-pass algorithm
- Min-max algorithm
- Delete term operation

4.12   User Interfaces for Analyses

The user interfaces for performing different types of analyses are shown in the workspaces menu (Figure 39).
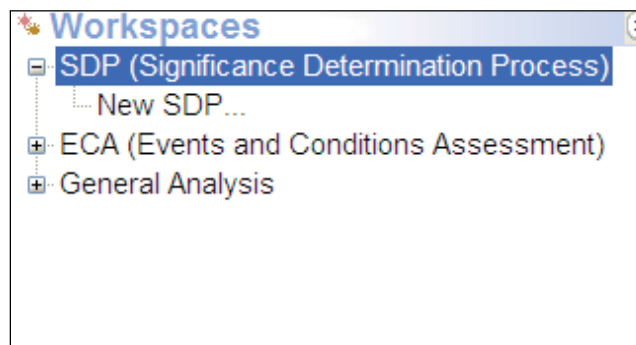

Figure 39. Workspaces window

In SAPHIRE 8, the act of creating a model has been separated from the analysis of that model in order to improve the quality of both the model (e.g., by avoiding inadvertent changes) and the analysis.

Consequently, in SAPHIRE 8, the analysis of models is performed by using what are called Workspaces. Currently, there are Workspaces for three types of analyses:

1. The NRC's Accident Sequence Precursor program, where the workspace is called "Events and Condition Assessment (ECA)"

2. The NRC's Significance Determination Process (SDP)

3. The General Analysis (GA) workspace

A model not identified as a SPAR model will only show the option for a General Analysis, which is a full option analysis. A model identified as a SPAR model (under Project → Modify) will also show the SDP and the ECA user interfaces, which are tailored analyses for specific applications.

Workspaces for each user interface are created and saved separately from the base model which keeps the original database intact. Each workspace created can have multiple analyses performed and saved. Workspaces are independent of each other and modifications or calculations made within one workspace will not affect another. Additional details on Workspaces may be found in Volume 5 of this NUREG/CR report.

Change sets are created in the workspaces by checking the desired checkboxes for systems, components, etc. These change sets are defined entirely within the selected workspace. A current limitation of workspaces is that changes sets which are defined in the SA user interface are not automatically carried over into the workspace when the model is loaded into a workspace from the SA space. If it is desired to apply such change sets, they would need to be mapped manually into the workspace.

5. CONCLUSIONS

SAPHIRE can be used to model a complex system's response to initiating events and quantify associated consequential outcome frequencies (or probabilities). Specifically, for nuclear power plant applications, SAPHIRE 8 can identify important contributors to core damage (Level 1 PRA) and containment failure during a severe accident which leads to releases (Level 2 PRA). It can be used for a PRA where the reactor is at full power, low power, or at shutdown conditions. Furthermore, it can be used to analyze both internal and external initiating events and has special features for managing models such as flooding and fire. It can also be used in a limited manner to quantify risk, using PRA techniques, in terms of release consequences to the public and environment (Level 3 PRA).

In SAPHIRE 8, the act of creating a model has been separated from the analysis of that model in order to improve the quality of both the model (e.g., by avoiding inadvertent changes) and the analysis. Consequently, in SAPHIRE 8, the analysis of models is performed by using what are called Workspaces. Currently, there are Workspaces for three types of analyses: (1) the NRC's Accident Sequence Precursor program, where the workspace is called "Events and Condition Assessment (ECA);" (2) the NRC's Significance Determination Process (SDP); and (3) the General Analysis (GA) workspace. Workspaces are independent of each other and modifications or calculations made within one workspace will not affect another. In addition, each workspace has a user interface and reports tailored for their intended uses.

This report provided an overview of the functions and features available in SAPHIRE 8 and presents general instructions for using the software. Since SAPHIRE 8 expands upon Version 7, new and improved features were discussed, while still referring to older options that have been carried forward into the SAPHIRE 8 version.

U.S. NUCLEAR REGULATORY COMMISSION

BIBLIOGRAPHIC DATA SHEET

(See Instructions on the reverse)

1. REPORT NUMBER (Assigned by NRC, Add Vol., Supp., Rev., and Addendum Numbers, if any.)
NUREG/CR-7039, Vol. 1
INL/EXT-09-17009

2. TITLE AND SUBTITLE

Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE) Version 8

Volume 1: Summary and Overview

3. DATE REPORT PUBLISHED	
MONTH	YEAR
MARCH	2011

4. FIN OR GRANT NUMBER
N6423

5. AUTHOR(S)

C. L. Smith, S. T. Wood

6. TYPE OF REPORT
Technical

7. PERIOD COVERED (Inclusive Dates)

8. PERFORMING ORGANIZATION - NAME AND ADDRESS (If NRC, provide Division, Office or Region, U.S. Nuclear Regulatory Commission, and mailing address; if contractor, provide name and mailing address.)

Idaho National Laboratory
Battelle Energy Alliance
P.O. Box 1625
Idaho Falls, ID 83415-3850

9. SPONSORING ORGANIZATION - NAME AND ADDRESS (If NRC, type "Same as above"; If contractor, provide NRC Division, Office or Region, U.S. Nuclear Regulatory Commission, and mailing address.)

Division of Risk Analysis
Office of Nuclear Regulatory Research
U.S. Nuclear Regulatory Commission
Washington, DC 20555-0001

10. SUPPLEMENTARY NOTES

D. O'Neal, NRC Project Manager

11. ABSTRACT (200 words or less)

The Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE) is a software application developed for performing a complete probabilistic risk assessment (PRA) using a personal computer (PC) running the Microsoft Windows operating system. SAPHIRE Version 8 is funded by the U.S. Nuclear Regulatory Commission (NRC) and developed by the Idaho National Laboratory (INL). INL's primary role in this project is that of software developer and tester. However, INL also plays an important role in technology transfer by interfacing and supporting SAPHIRE users, who constitute a wide range of PRA practitioners from the NRC, national laboratories, the private sector, and foreign countries. This report provides an overview of the functions and features available in SAPHIRE 8 and presents general instructions for using the software. Since SAPHIRE 8 expands upon Version 7, new and improved features will be discussed.

12. KEY WORDS/DESCRIPTORS (List words or phrases that will assist researchers in locating the report.)

SAPHIRE 8, software, reliability, risk, safety, PRA

13. AVAILABILITY STATEMENT
Unlimited

14. SECURITY CLASSIFICATION
(This page)
Unclassified
(This report)
Unclassified

15. NUMBER OF PAGES

16. PRICE